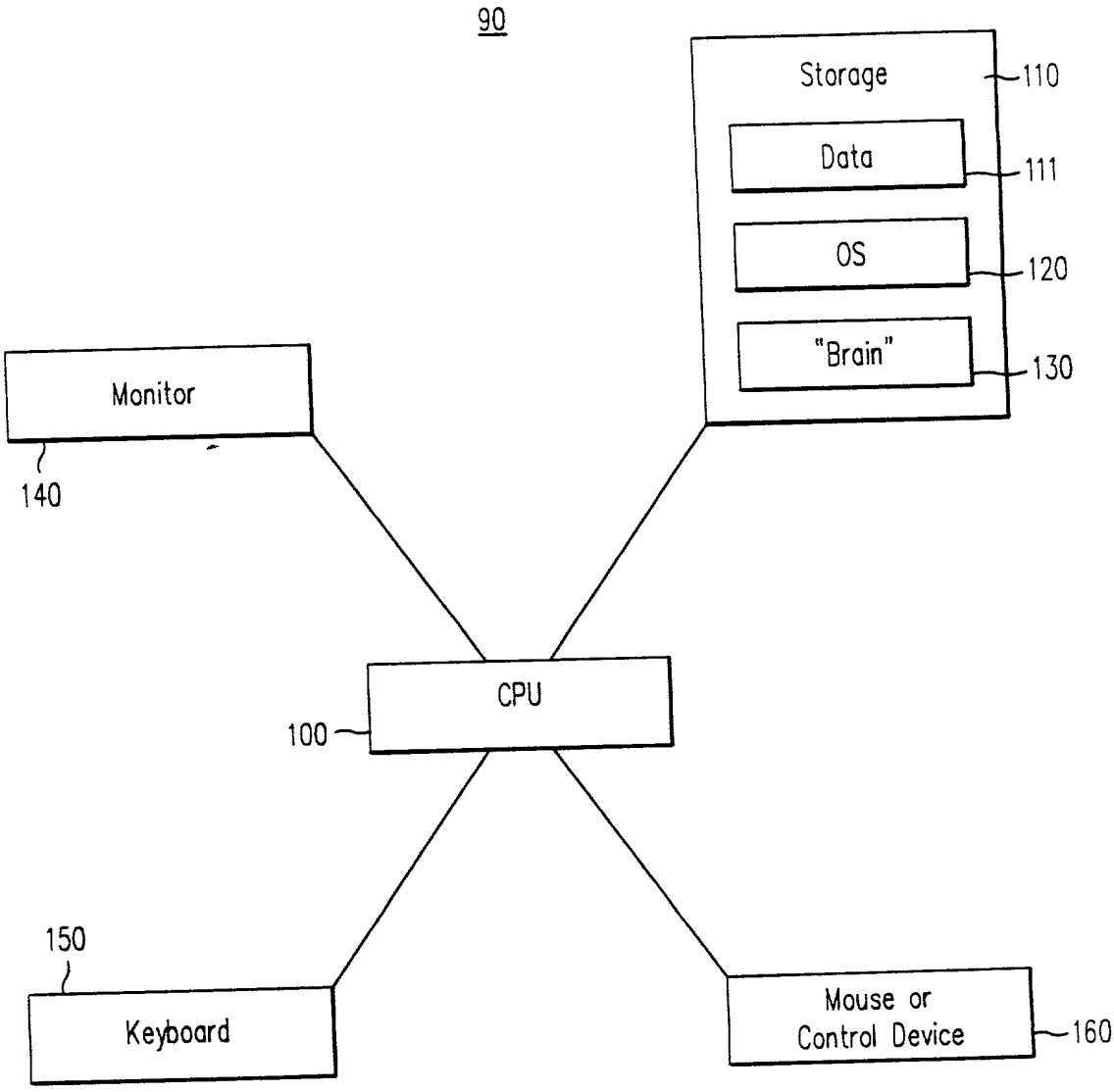

90
Storage
110
Data
111
OS
120
"Brain"
130
Monitor
140
CPU
100
150
Keyboard
Mouse or
Control Device
160

FIG. 1

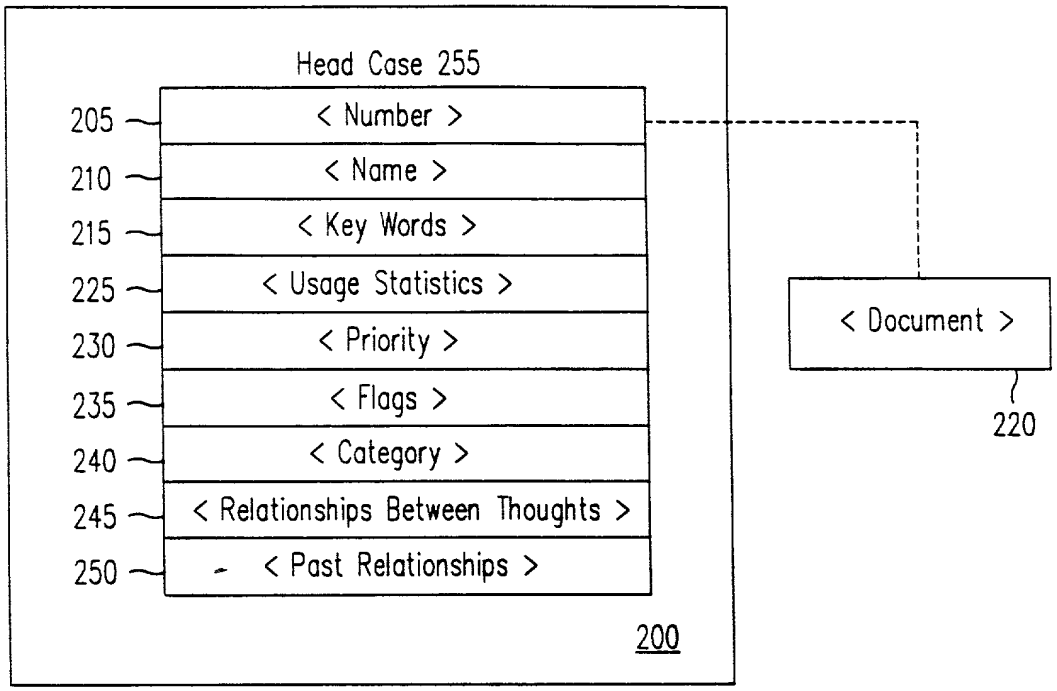

Head Case 255
205
< Number >
210
< Name >
215
< Key Words >
225
< Usage Statistics >
230
< Priority >
235
< Flags >
240
< Category >
245
< Relationships Between Thoughts >
250
< Past Relationships >
200
< Document >
220

FIG. 2

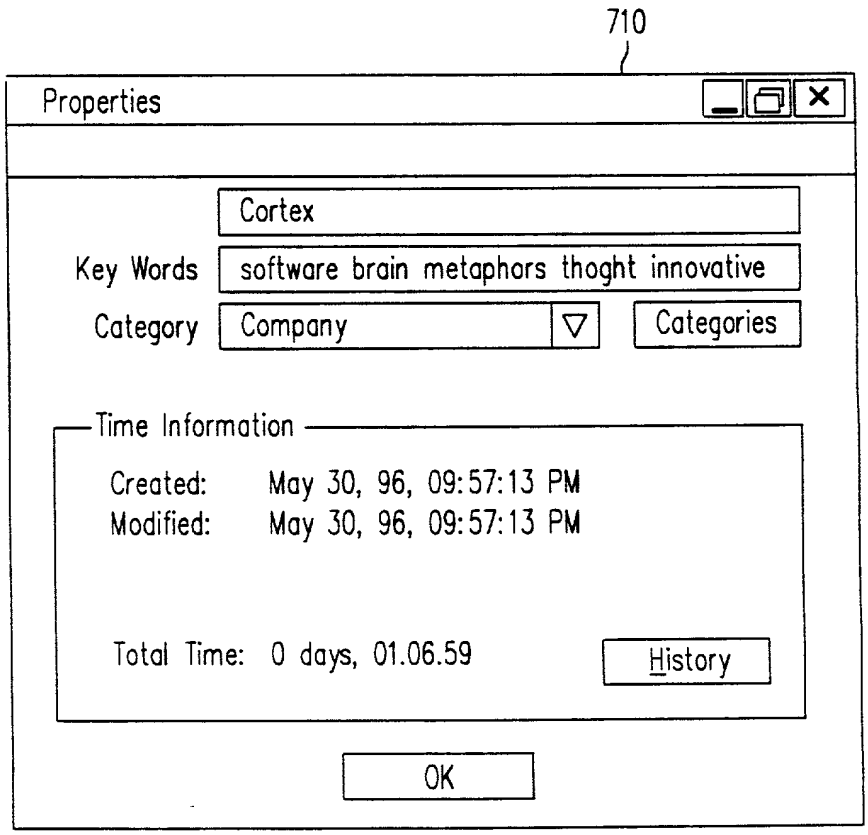

710
Properties
Cortex
Key Words
software brain metaphors thoght innovative
Category
Company
Categories
Time Information
Created: May 30, 96, 09:57:13 PM
Modified: May 30, 96, 09:57:13 PM
Total Time: 0 days, 01.06.59
History
OK

FIG. 7

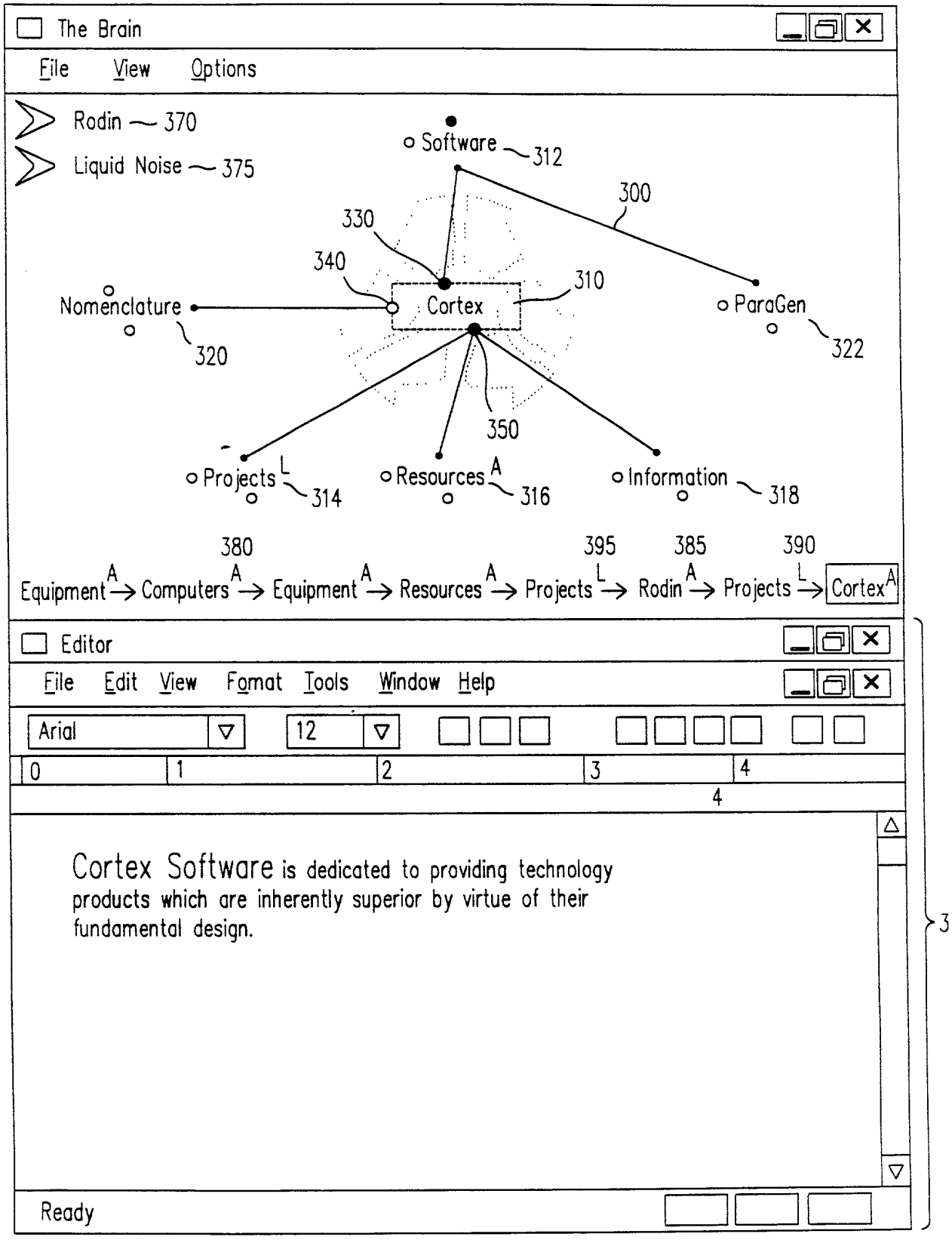

The Brain
File View Options
Rodin 370
Liquid Noise 375
Software 312
300
330
340
310
Nomenclature
320
Cortex
ParaGen
322
350
Projects L
314
Resources A
316
Information
318
380 395 385 390
Equipment A → Computers A → Equipment A → Resources A → Projects L → Rodin A → Projects L → Cortex A
Editor
File Edit View Format Tools Window Help
Arial
12
0 1 2 3 4
4
Cortex Software is dedicated to providing technology products which are inherently superior by virtue of their fundamental design.
Ready
3

FIG. 3

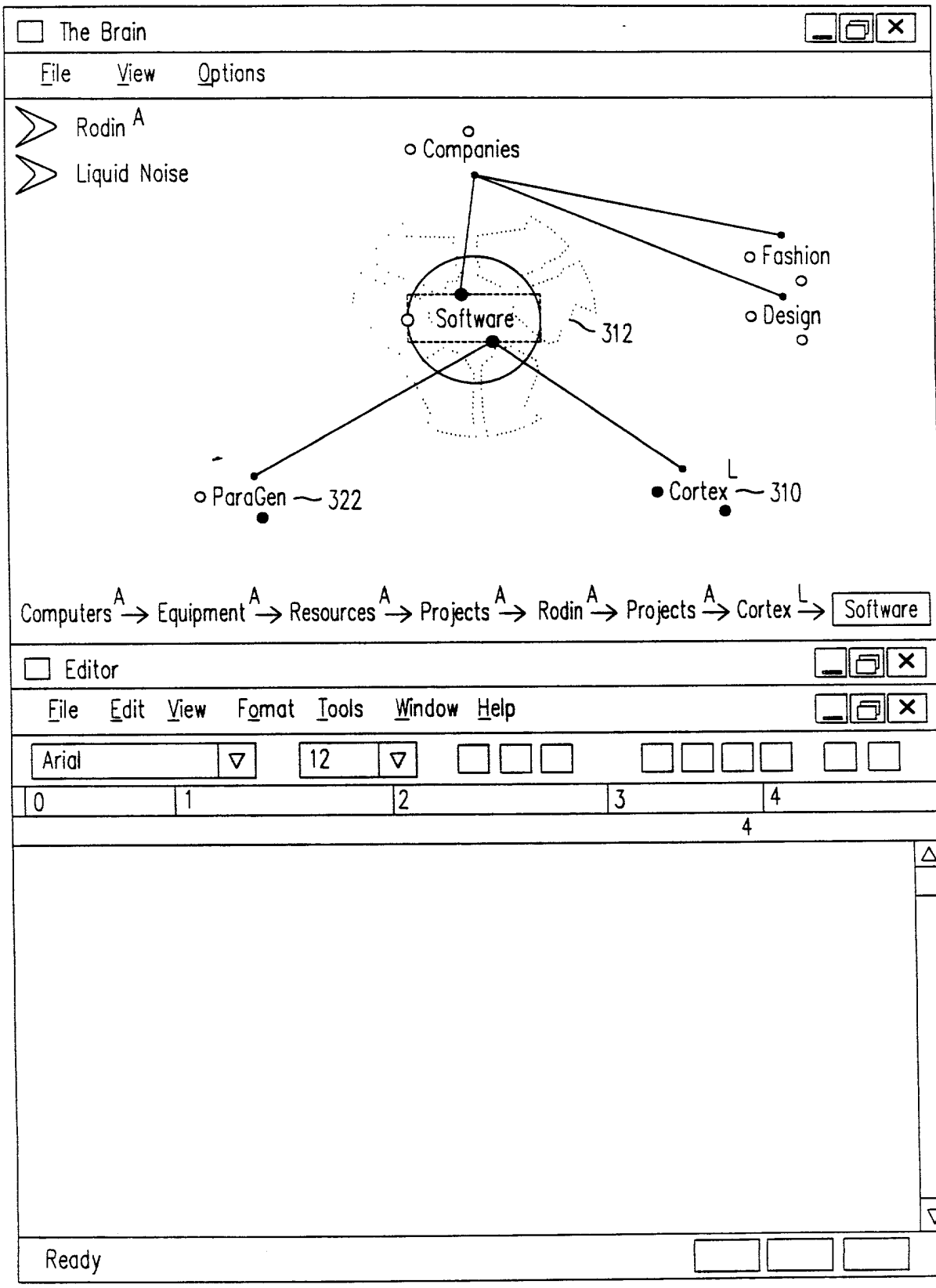

The Brain
File View Options
Rodin A
Liquid Noise
Companies
Fashion
Design
Software
312
ParaGen 322
Cortex L 310
Computers A → Equipment A → Resources A → Projects A → Rodin A → Projects A → Cortex L → Software
Editor
File Edit View Format Tools Window Help
Arial
12
0 1 2 3 4
4
Ready

FIG. 4

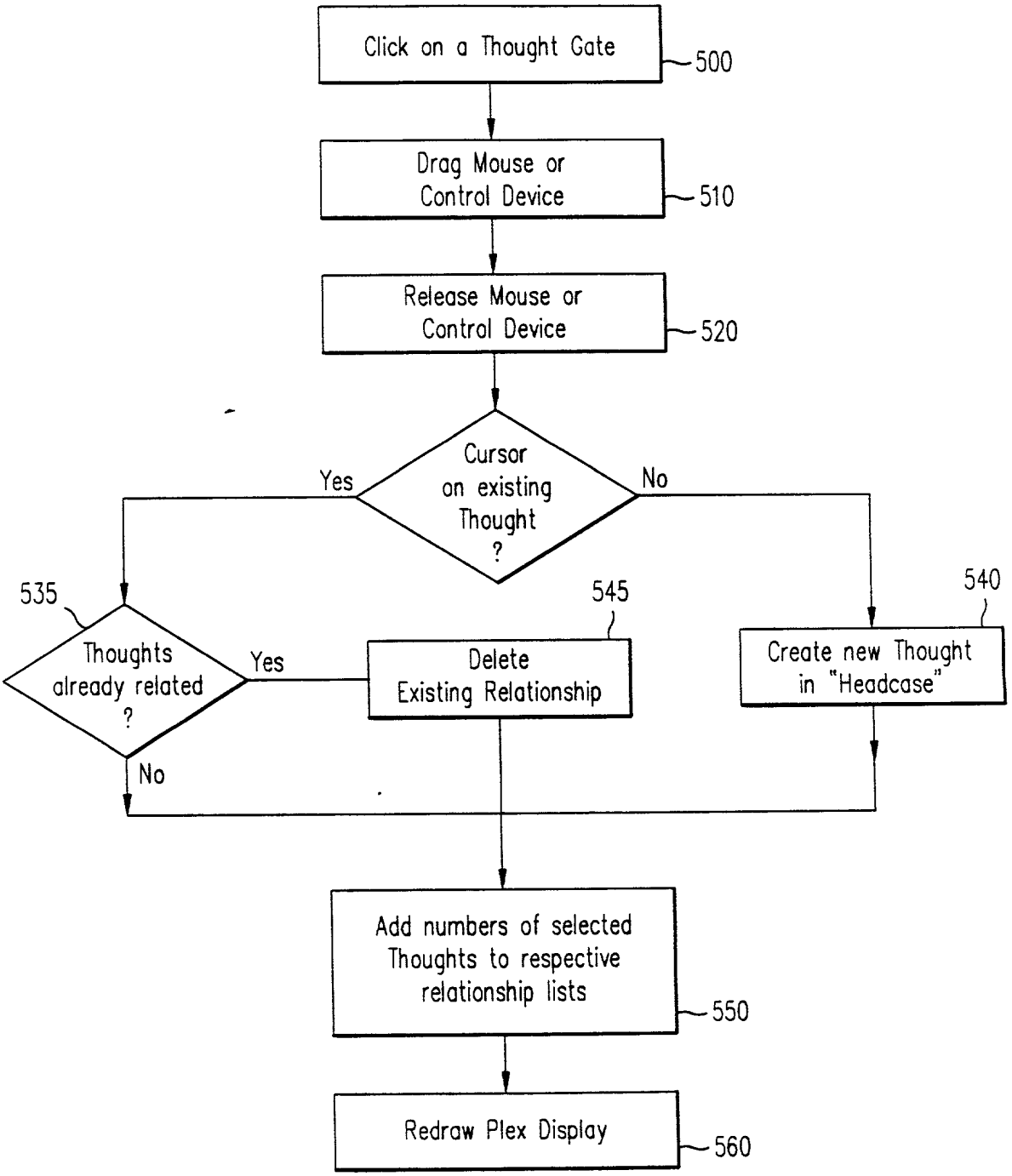

Click on a Thought Gate
500
Drag Mouse or Control Device
510
Release Mouse or Control Device
520
Cursor on existing Thought ?
Yes
No
535
Thoughts already related ?
Yes
No
545
Delete Existing Relationship
540
Create new Thought in "Headcase"
Add numbers of selected Thoughts to respective relationship lists
550
Redraw Plex Display
560

FIG. 5

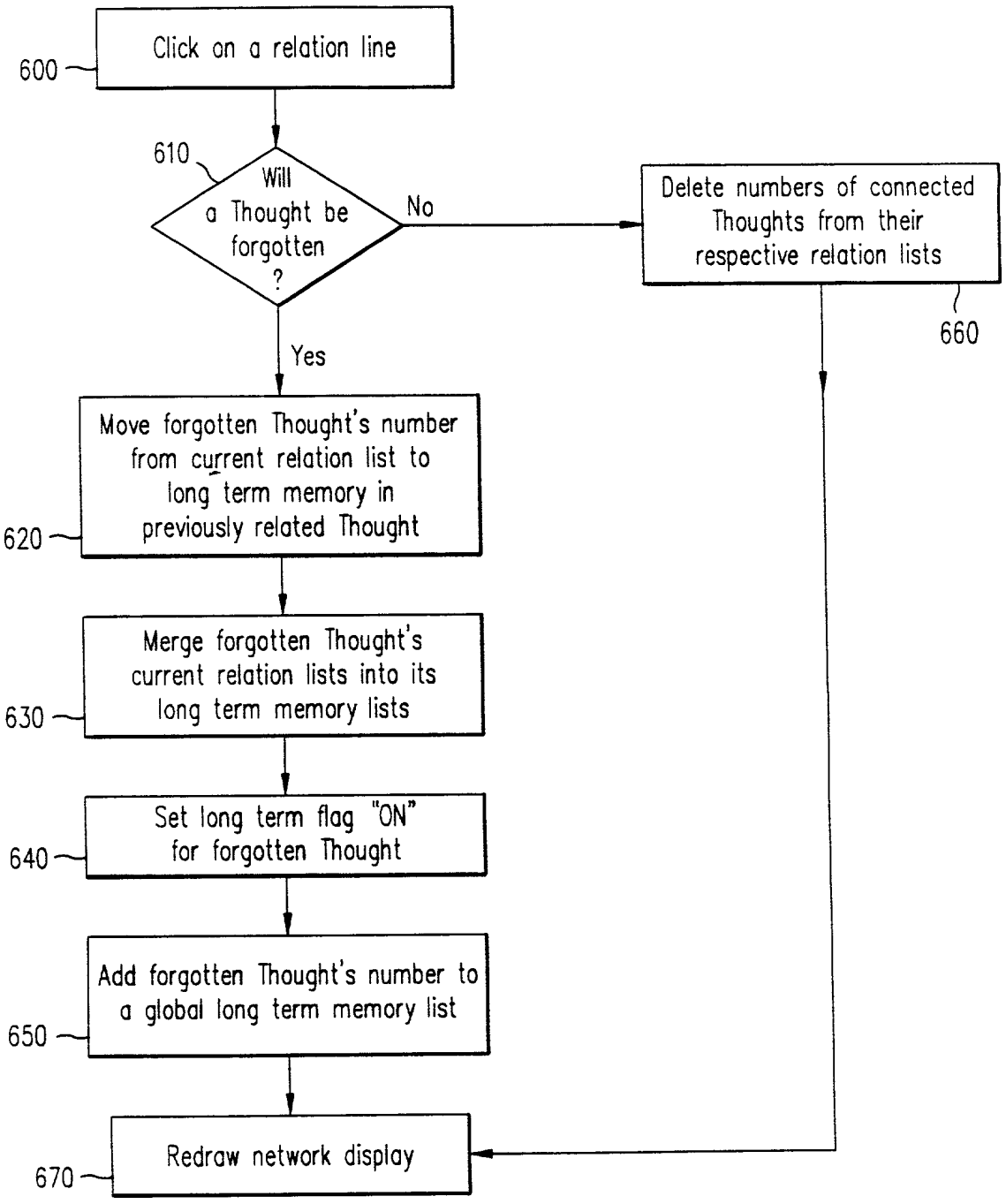

600
Click on a relation line
610
Will a Thought be forgotten ?
No
Delete numbers of connected Thoughts from their respective relation lists
660
Yes
620
Move forgotten Thought's number from current relation list to long term memory in previously related Thought
630
Merge forgotten Thought's current relation lists into its long term memory lists
640
Set long term flag "ON" for forgotten Thought
650
Add forgotten Thought's number to a global long term memory list
670
Redraw network display

FIG. 6

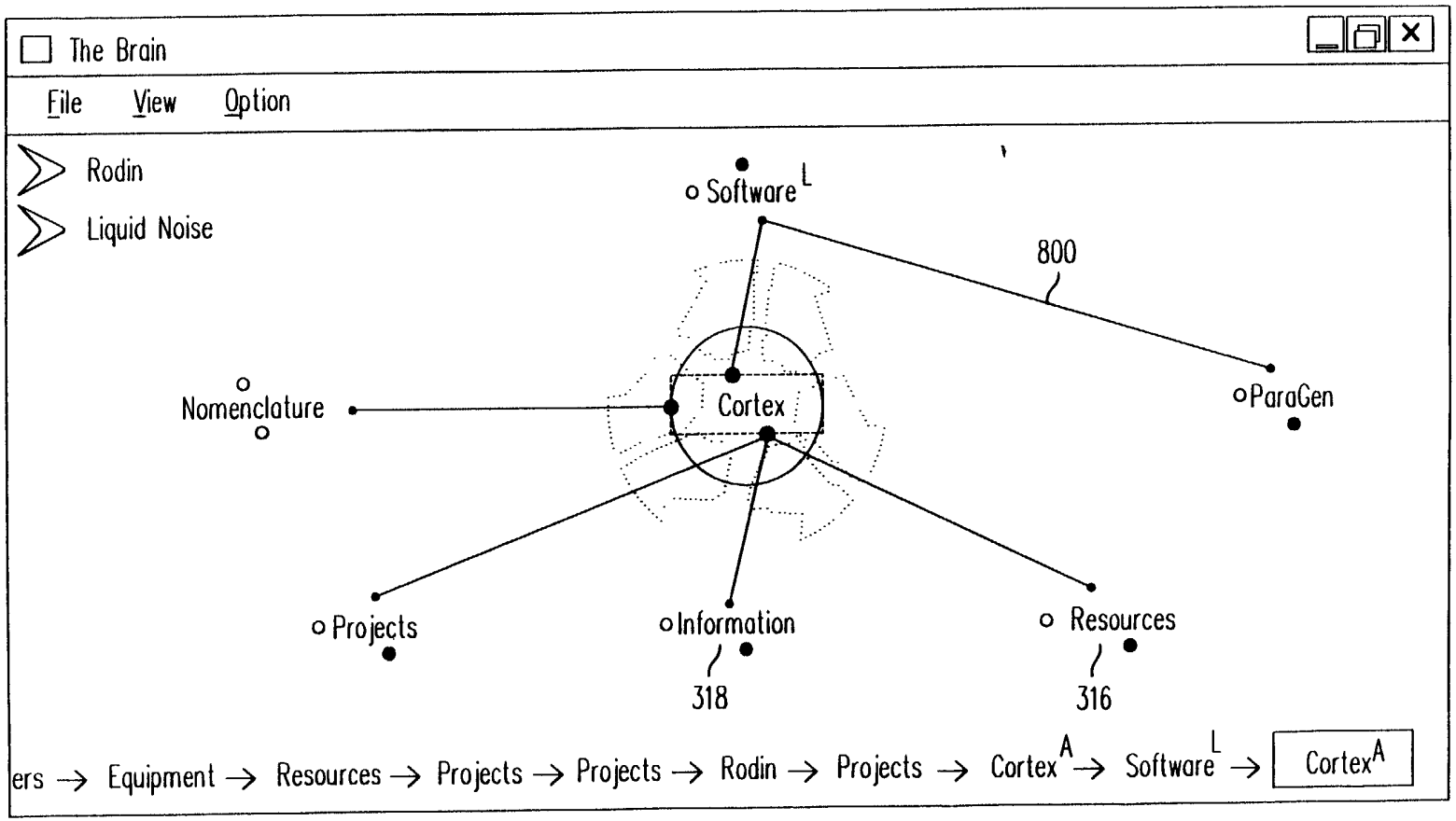

The Brain
File
View
Option
Rodin
Liquid Noise
Software L
800
Nomenclature
Cortex
ParaGen
Projects
Information
Resources
318
316
ers → Equipment → Resources → Projects → Projects → Rodin → Projects → Cortex A → Software L → Cortex A

FIG. 8

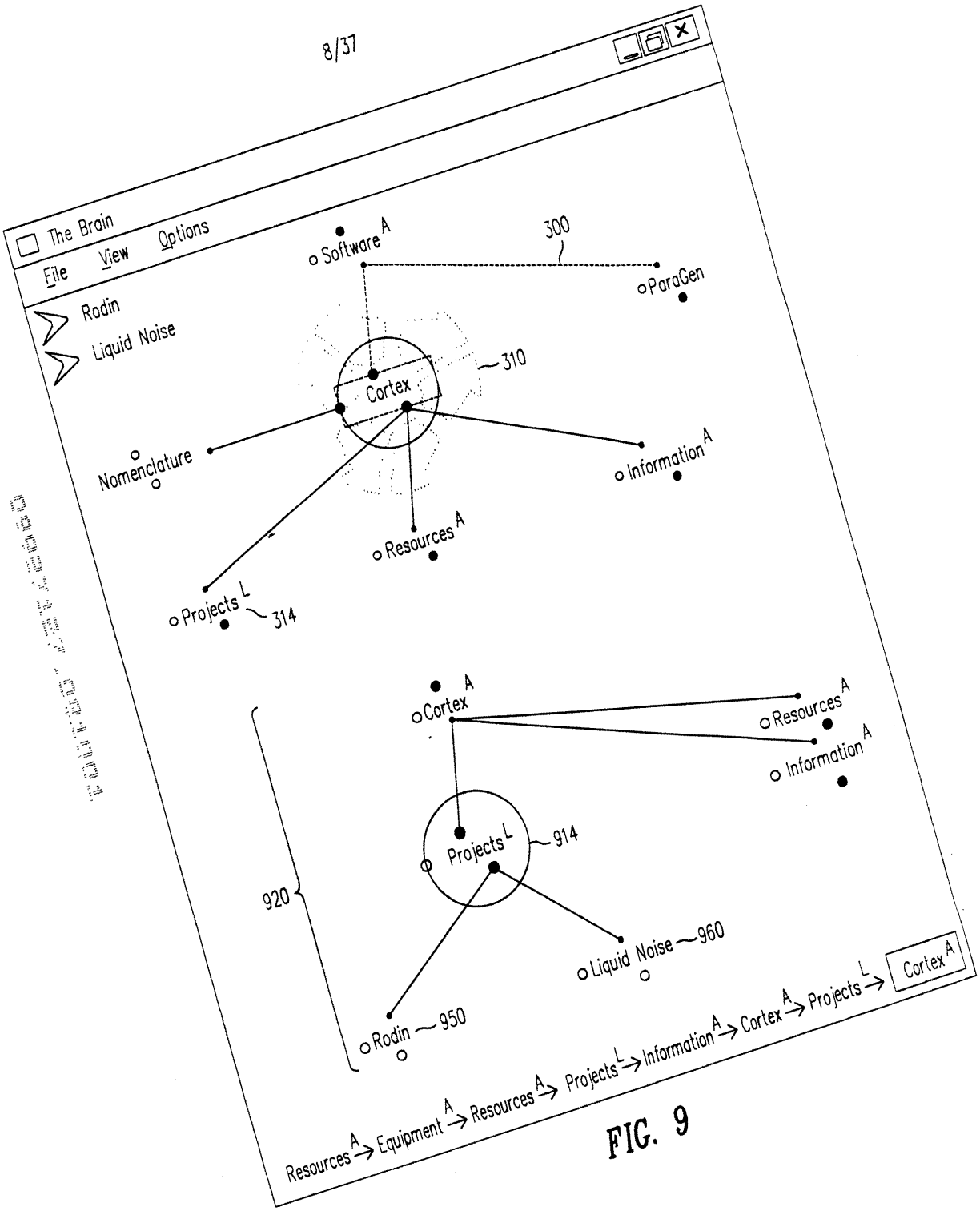

The Brain
File View Options
Radin
Liquid Noise
Software A
300
ParaGen
Cortex
310
Nomenclature
Information A
Resources A
Projects L
314
Cortex A
Resources A
Information A
Projects L
914
920
Liquid Noise 960
Radin 950
Resources A → Equipment A → Resources A → Projects L → Information A → Cortex A → Projects L → Cortex A

FIG. 9

```
boolean CheckForIsolation(int centralThought, int targetThought)
{
      //this function checks if centralThought is related to targetThought
      //via any of targetThought's relations (not directly)

      //remove centralThought as a direct relation from targetThought
      RemoveRelation(targetThought, centralThought);

      //create an empty thought list to keep track of the search
        intList searchList=CreateEmptyList();

      //start recursive searches on each of targetThought's direct relations
        int relation=GetFirstRelation(targetThought);
        boolean found;
        do {
                found=Search(relation, centralThought, searchList);
                if(found) {
                        //centralThought was found, no need to search any further
                        break;
                }
               //this loop will end when there are no more relations
       } while(relation=GetNextRelation(targetThought);

      //add centralThought back onto targetThought as a relation
      AddRelation(targetThought, centralThought);

      return found;
}
```

FIG. 10a

Figure 10, cont'd

```
boolean Search(source, dest, seachList)
{
        if(Find(source, searchList)) {
              //source has already been searched
              return FALSE;
        }

        //add source to the searchList
        Add(source, searchList)

        if(source == dest) {
              //this is the destination,we have found it
              return TRUE;
        }

        //recursive searches on each of sources direct relations
        int relation=GetFirstRelation(source);
        boolean found;
        do {
              found=Search(relation, dest, searchList);
              if(found) {
                    //centralThought was found, no need to search any further
                    break;
              }
             //this loop will end when there are no more relations
        } while(relation=GetNextRelation(targetThought);

            return found;
}
```

FIG. 10b

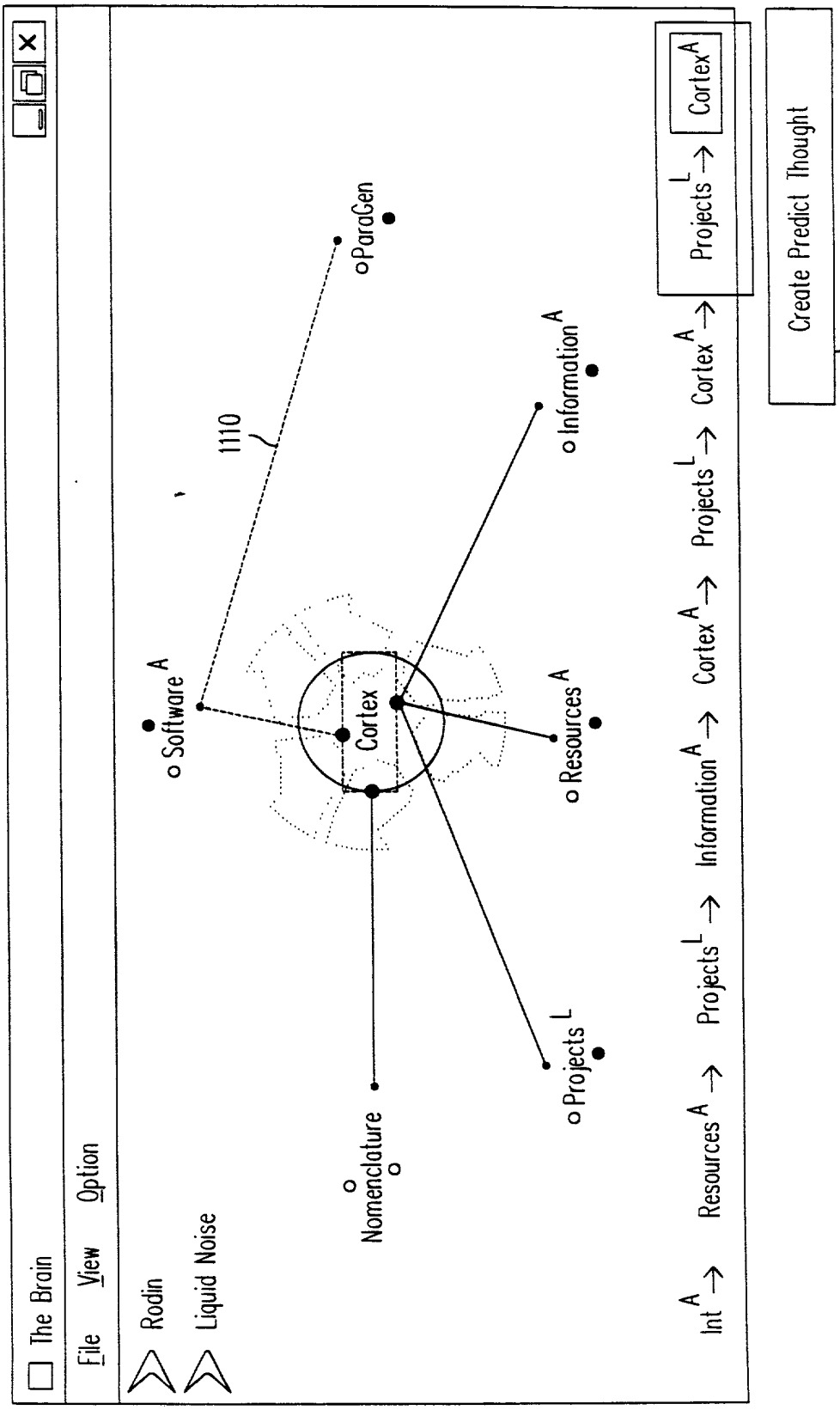

The Brain
File View Option
Rodin
Liquid Noise
Software A
ParaGen
1110
Cortex
Information A
Resources A
Projects L
Nomenclature
Int A → Resources A → Projects L → Information A → Cortex A → Projects L → Cortex A → Projects L → Cortex A
Create Predict Thought
1120

FIG. 11

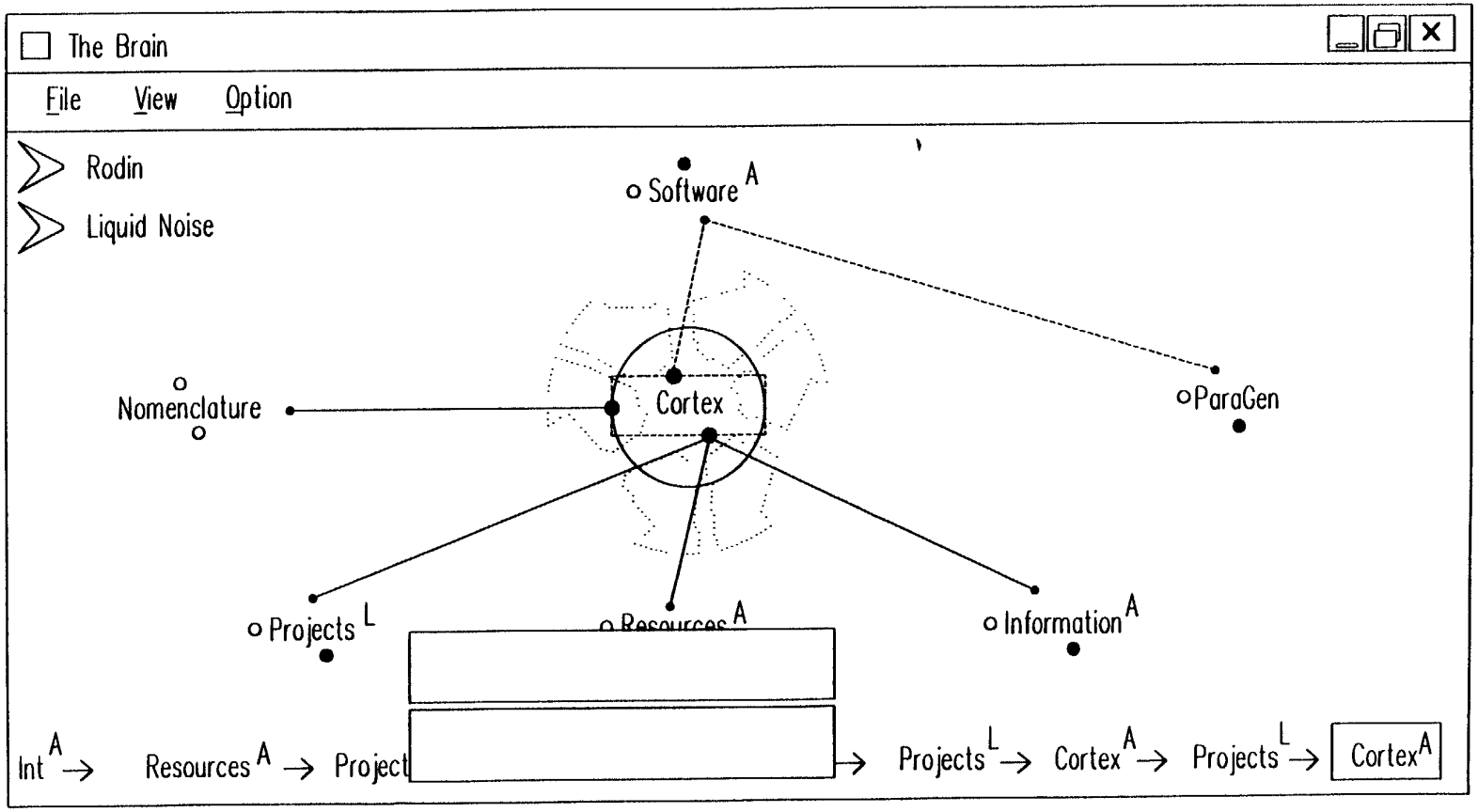

The Brain
File
View
Option
Rodin
Liquid Noise
Software A
Nomenclature
Cortex
ParaGen
Projects L
Resources A
Information A
Int A →
Resources A → Project
Projects L → Cortex A → Projects L →
Cortex A

FIG. 12

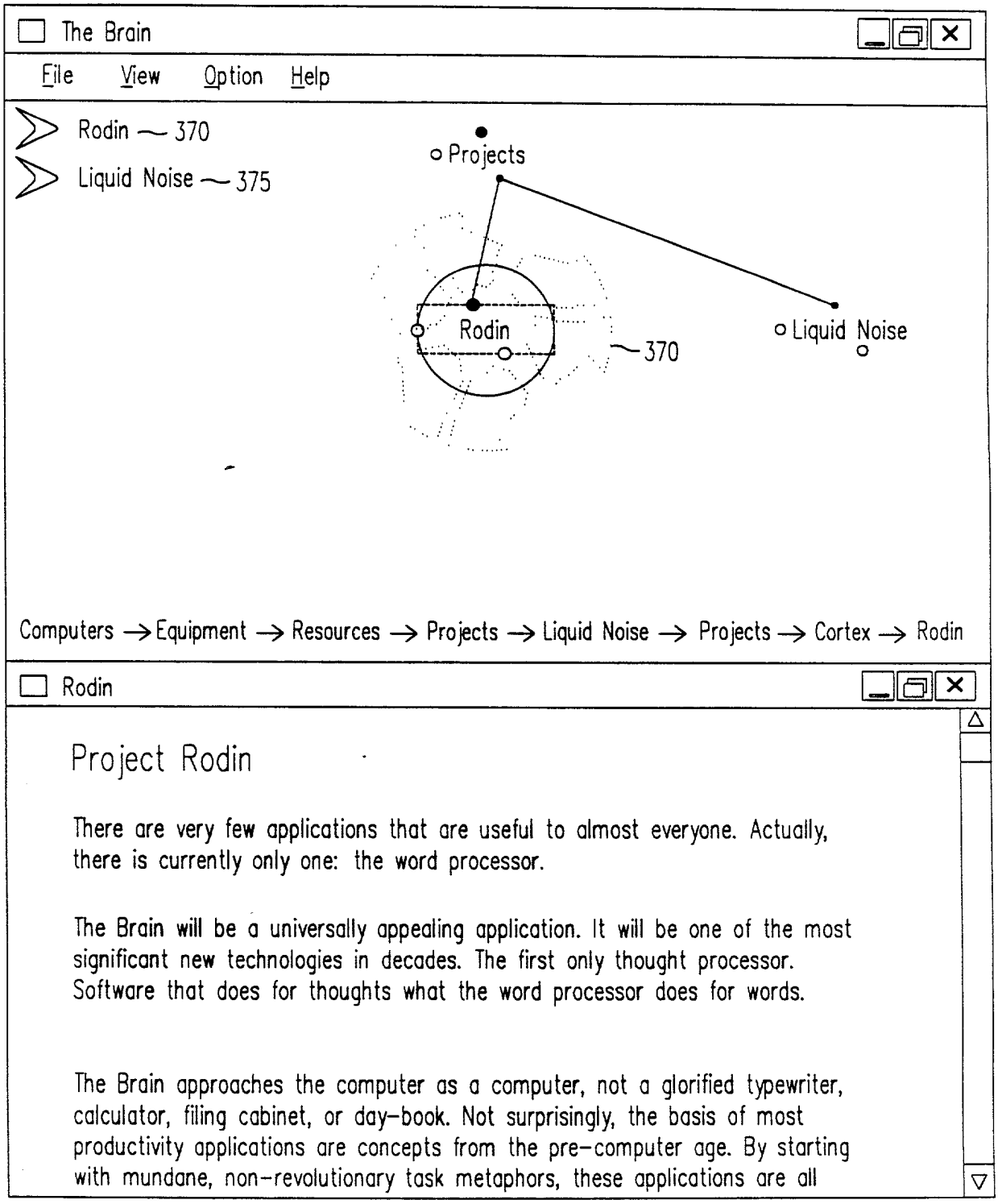

The Brain
File
View
Option
Help
Rodin — 370
Liquid Noise — 375
Projects
Rodin
370
Liquid Noise
Computers → Equipment → Resources → Projects → Liquid Noise → Projects → Cortex → Rodin
Rodin
Project Rodin
There are very few applications that are useful to almost everyone. Actually, there is currently only one: the word processor.
The Brain will be a universally appealing application. It will be one of the most significant new technologies in decades. The first only thought processor. Software that does for thoughts what the word processor does for words.
The Brain approaches the computer as a computer, not a glorified typewriter, calculator, filing cabinet, or day-book. Not surprisingly, the basis of most productivity applications are concepts from the pre-computer age. By starting with mundane, non-revolutionary task metaphors, these applications are all

FIG. 13

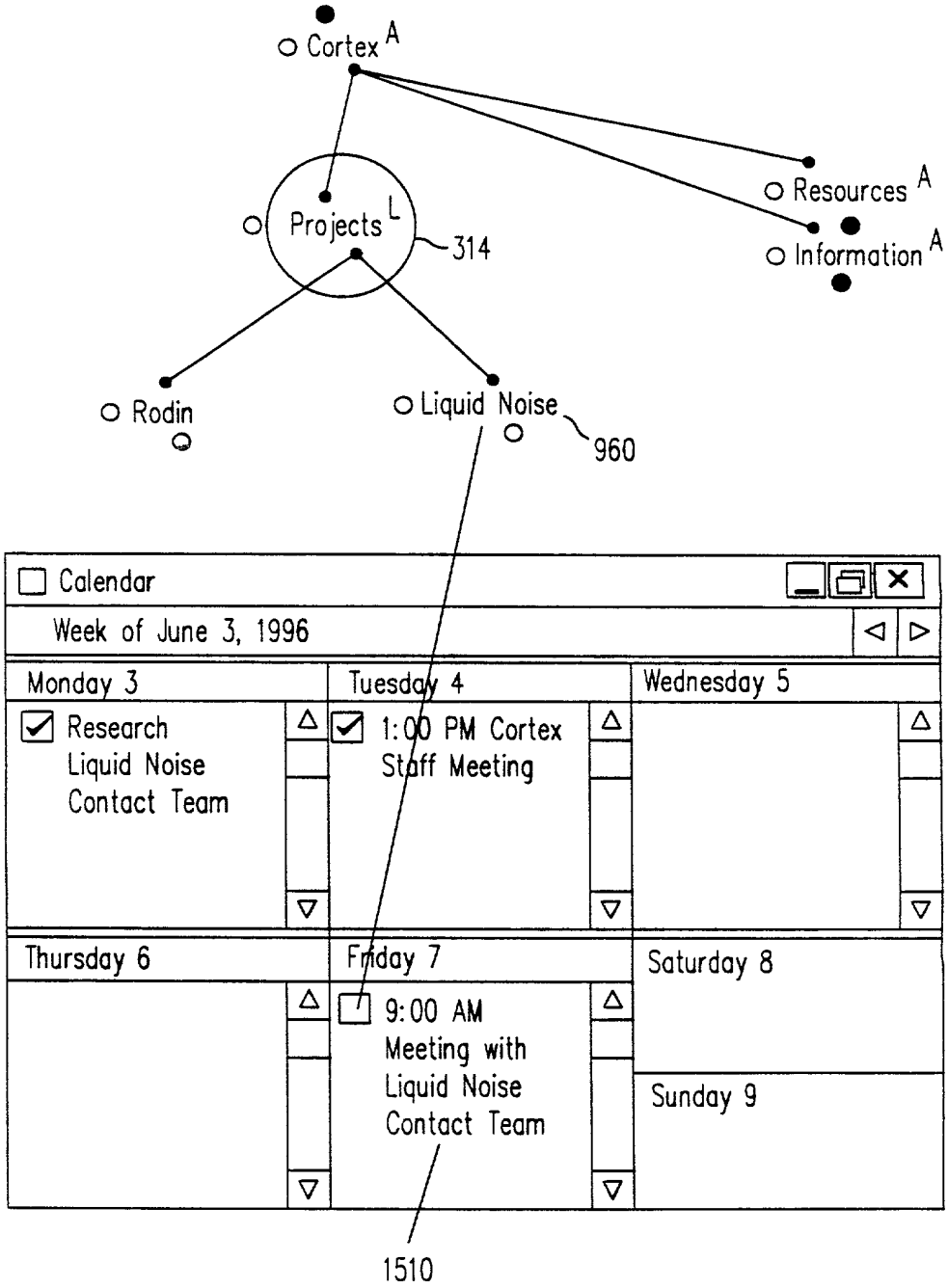

Cortex A
Resources A
Projects L
314
Information A
Rodin
Liquid Noise
960
Calendar
Week of June 3, 1996
Monday 3
Research
Liquid Noise
Contact Team
Tuesday 4
1:00 PM Cortex
Staff Meeting
Wednesday 5
Thursday 6
Friday 7
9:00 AM
Meeting with
Liquid Noise
Contact Team
Saturday 8
Sunday 9
1510

FIG. 15

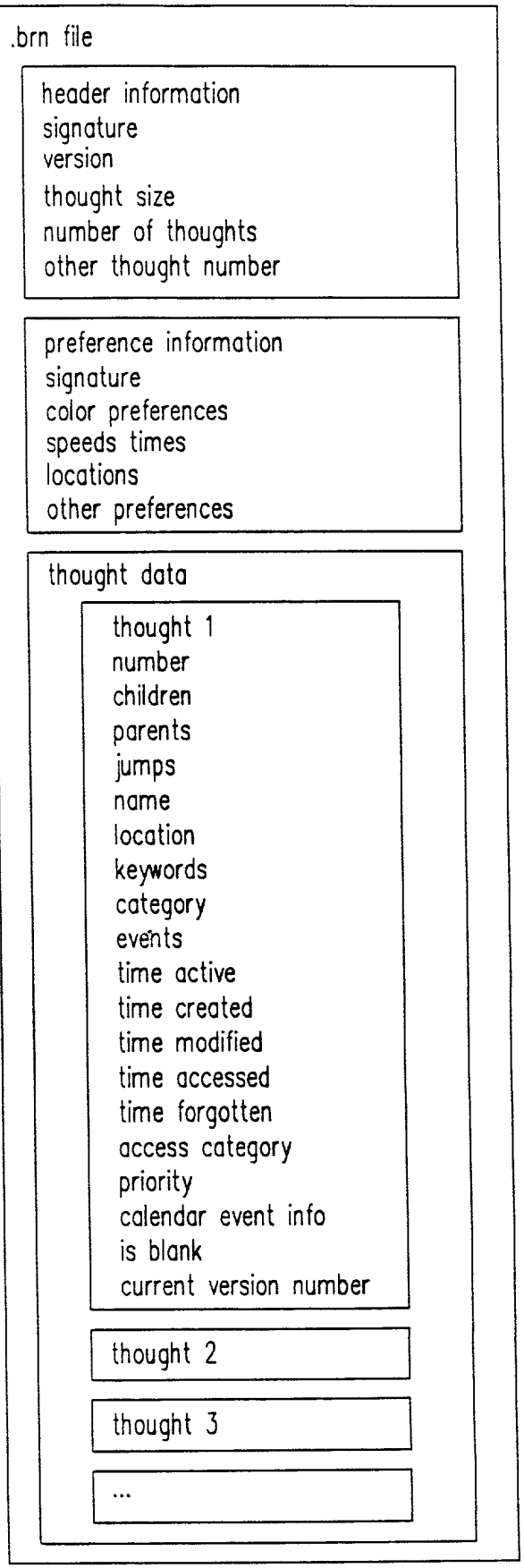

.brn file
header information
signature
version
thought size
number of thoughts
other thought number
preference information
signature
color preferences
speeds times
locations
other preferences
thought data
thought 1
number
children
parents
jumps
name
location
keywords
category
events
time active
time created
time modified
time accessed
time forgotten
access category
priority
calendar event info
is blank
current version number
thought 2
thought 3
...

FIG. 16

```
ForgetThought (fNum)
{
   //mark all the children of the selected thought
   list.Clear();
   MarkChildren (fNum, list);
   //unmark the active thought
   list, RemoveThought (activeThought);
   //unmark thoughts with unmarked parents
   lNum = list.GetFirstNum();
   while(lNum != 0)
   {
      if(lNum != fNum)    //don't unmark the selected thought
      {
       pNum = GetFirstThoughtParent (lNum);
       while(pNum != 0)
       {
         if(list.Contains(pNum) == FALSE)
         {
          if(IsThoughtInLongTermMemory (pNum) == FALSE)
           {
        //unmark all the children of the unmarked parent
        childList.Clear();

         MarkChildren(pNum, childList);
         list.RemoveList(childList);
        }
      }
      pNum = GetNextThoughtParent(lNum);
    }

  }
   lNum = list.GetNextNum();
  }
       //now forget all the thoughts left on the list
       lNum = list.GetFirstNum();
       while(lNum != 0)
      {
        ForgetThought(lNum);
        lNum = list.GetNextNum();
       }
     }
     RememberThought(rNum)
     {
        //mark all the children of the selected thought
        list.Clear();
     MarkChildren(rNum, list);
     // remember all the thoughts on the list
     lNum = list.GetFirstNum();
     while(lNum != 0)
     {
       RememberThought(lNum);
       lNum = list.GetNextNum();
     }
   }
   MarkChildren(num, list)
   {
     list.AddThought(num);
     cNum = GetFirstChild(num);
     while(cNum != 0)
     {
       MarkChildren(cNum, list);
       cNum = GetNextChild(num);
     }
}
```

FIG. 17

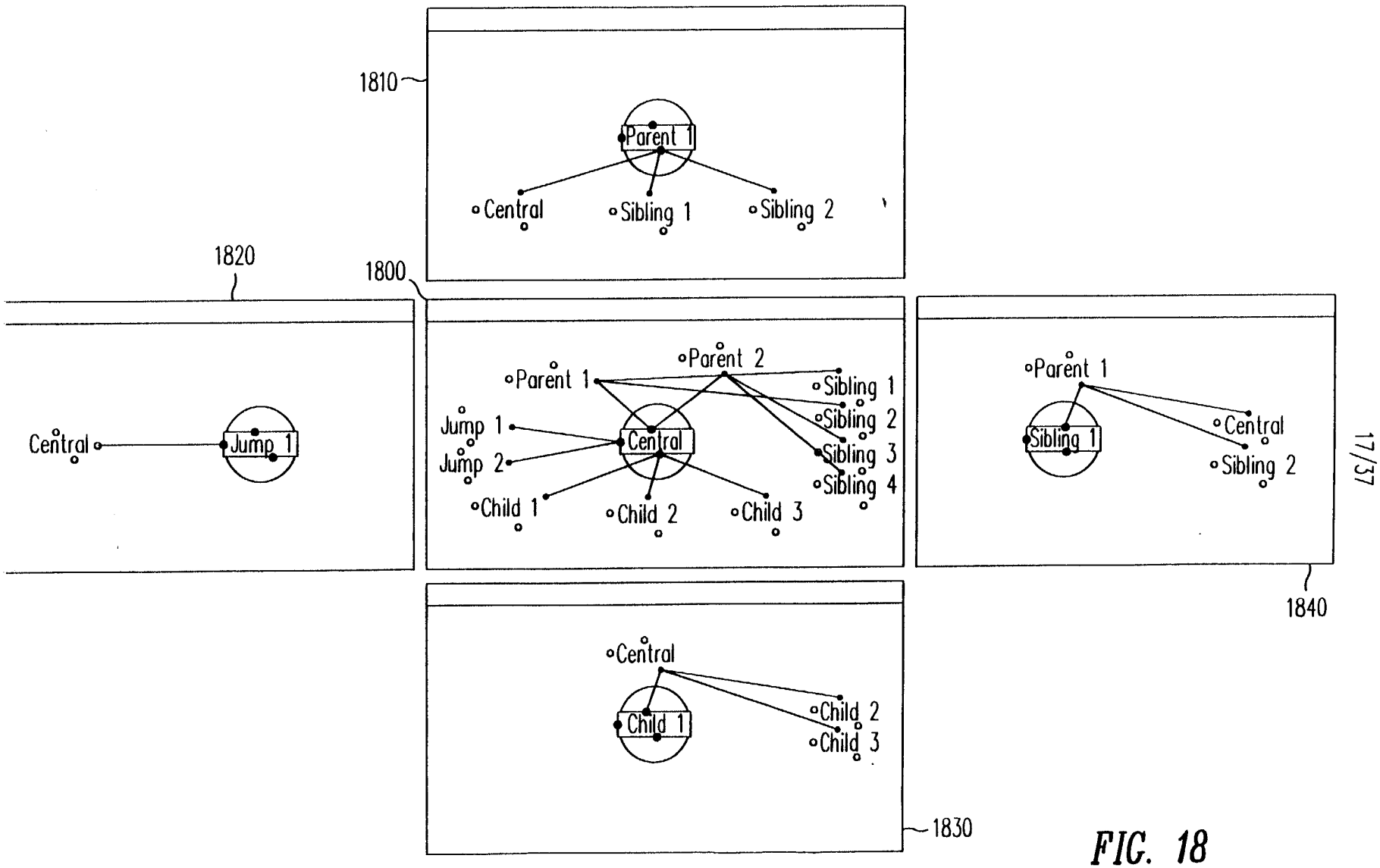

1810
Parent 1
Central
Sibling 1
Sibling 2
1820
1800
Central
Jump 1
Parent 1
Parent 2
Jump 1
Jump 2
Central
Child 1
Child 2
Child 3
Sibling 1
Sibling 2
Sibling 3
Sibling 4
Parent 1
Sibling 1
Central
Sibling 2
1840
Central
Child 1
Child 2
Child 3
1830

FIG. 18

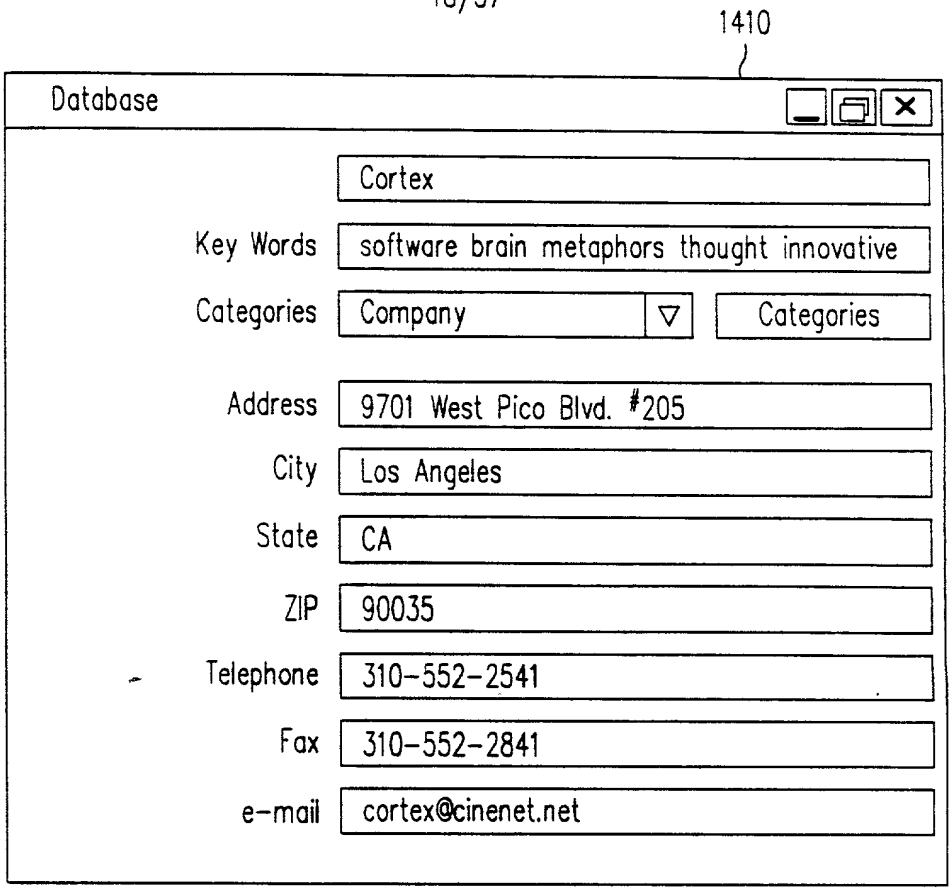

1410
Database
Cortex
Key Words
software brain metaphors thought innovative
Categories
Company
Categories
Address
9701 West Pico Blvd. #205
City
Los Angeles
State
CA
ZIP
90035
Telephone
310-552-2541
Fax
310-552-2841
e-mail
cortex@cinenet.net

FIG. 14

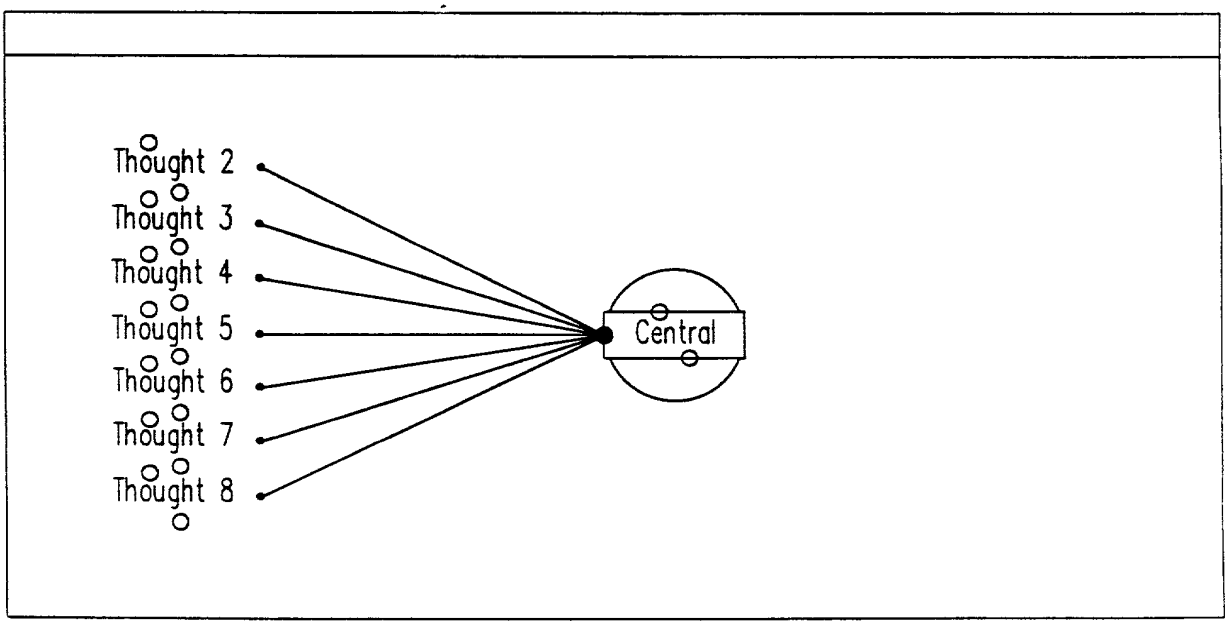

Thought 2
Thought 3
Thought 4
Thought 5
Thought 6
Thought 7
Thought 8
Central

FIG. 19

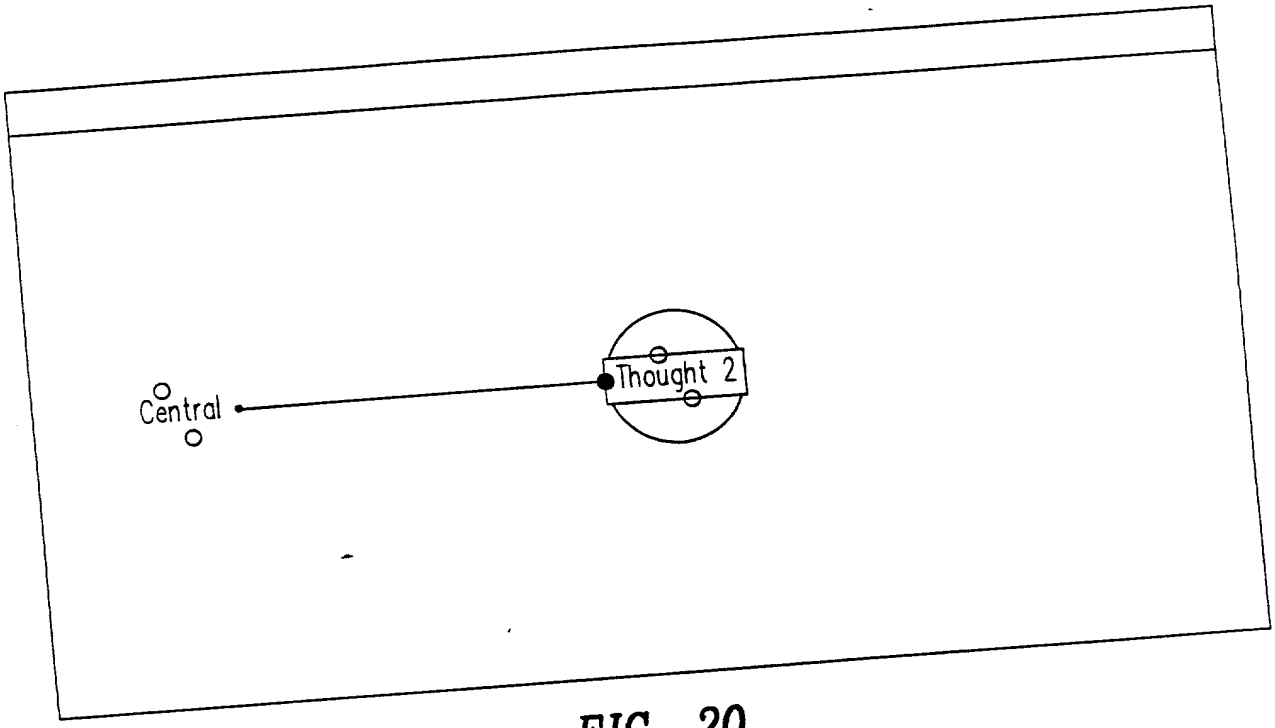

Central
Thought 2

FIG. 20

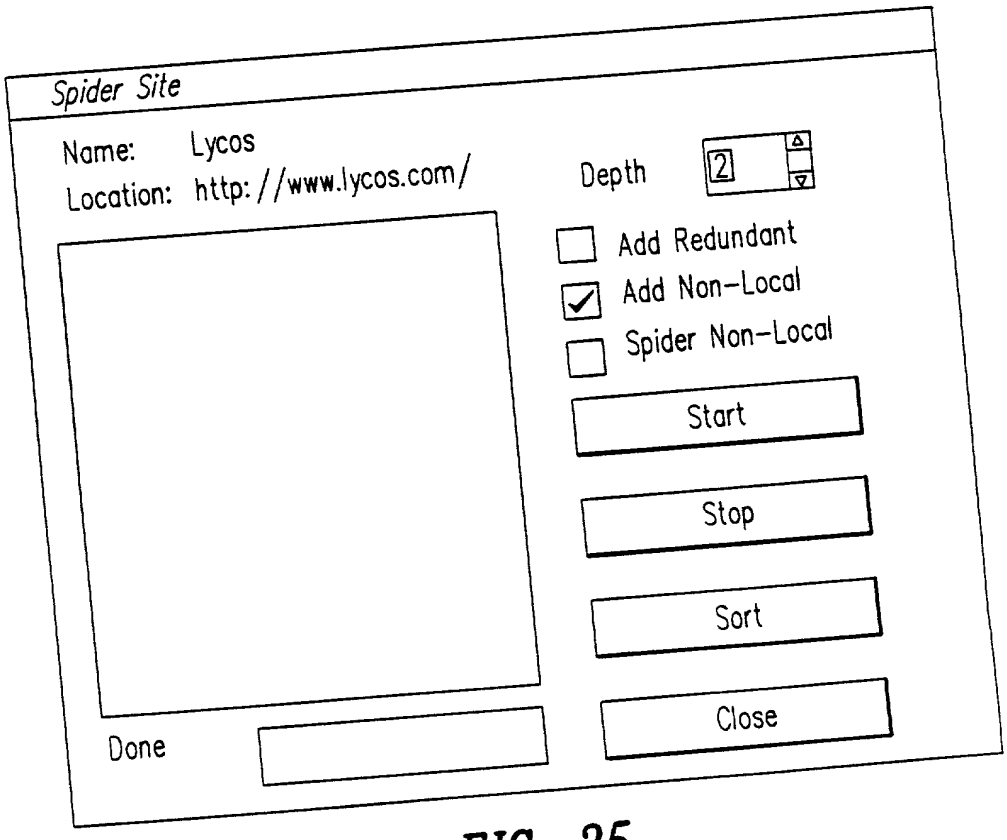

Spider Site
Name: Lycos
Location: http://www.lycos.com/
Depth
2
Add Redundant
Add Non-Local
Spider Non-Local
Start
Stop
Sort
Close
Done

FIG. 35

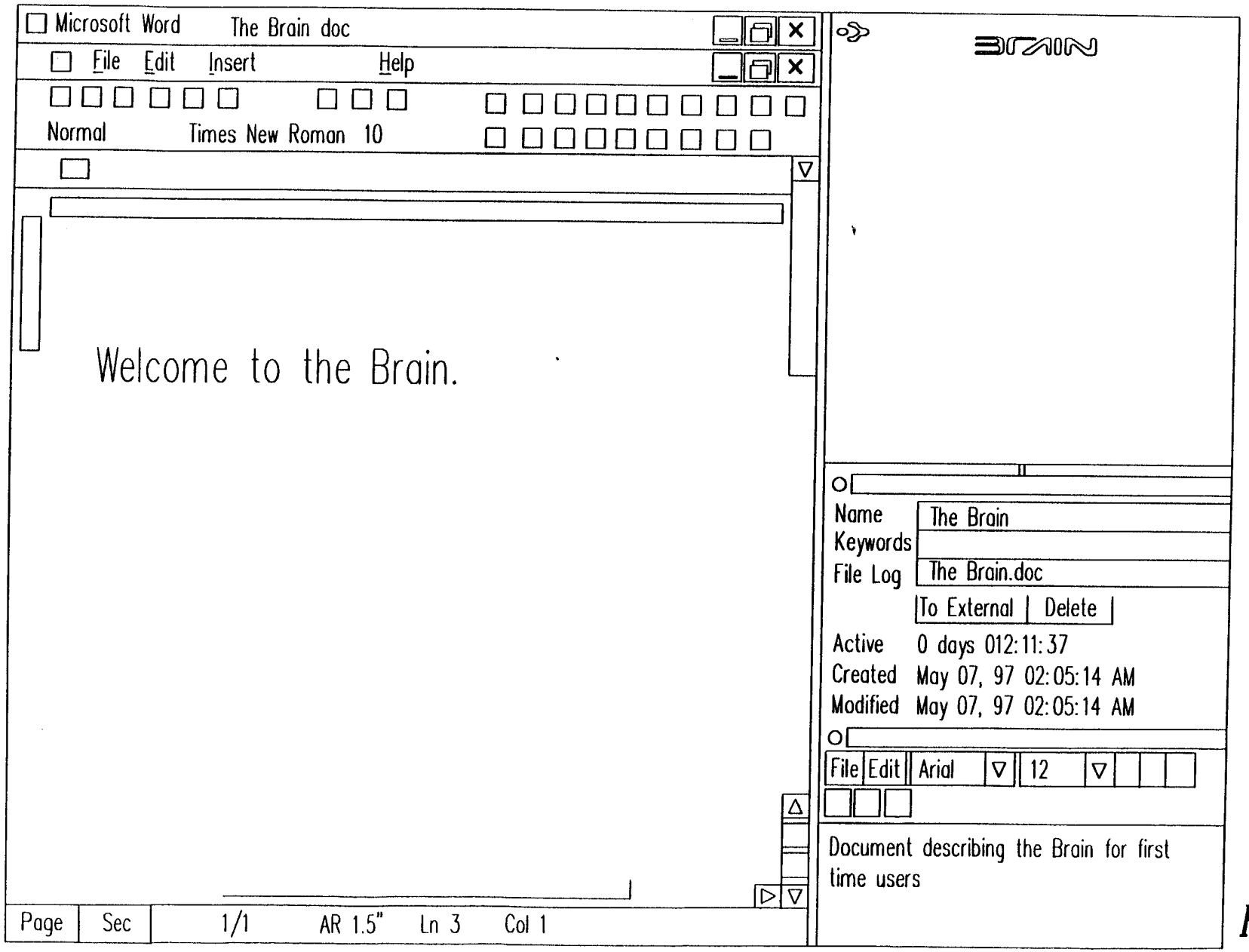

Microsoft Word
The Brain doc
File Edit Insert Help
Normal
Times New Roman 10
Welcome to the Brain.
Page
Sec
1/1
AR 1.5"
Ln 3
Col 1
BRAIN
Name
The Brain
Keywords
File Log
The Brain.doc
To External
Delete
Active
0 days 012:11:37
Created
May 07, 97 02:05:14 AM
Modified
May 07, 97 02:05:14 AM
File
Edit
Arial
12
Document describing the Brain for first time users

FIG. 21

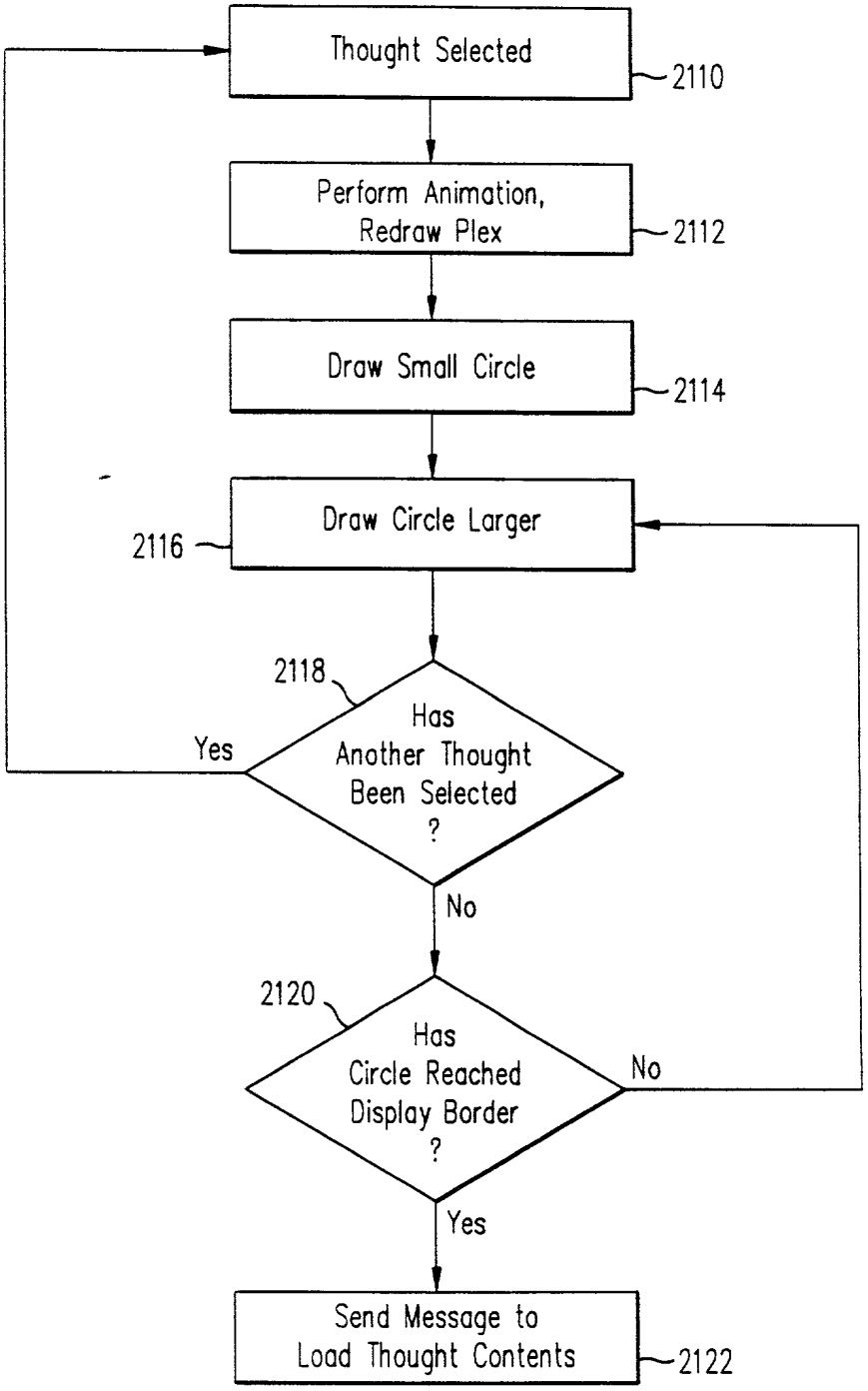

Thought Selected
2110
Perform Animation, Redraw Plex
2112
Draw Small Circle
2114
Draw Circle Larger
2116
2118
Has Another Thought Been Selected ?
Yes
No
2120
Has Circle Reached Display Border ?
No
Yes
Send Message to Load Thought Contents
2122

FIG. 22

Algorithm for drawing the plex with distant thoughts

```
1.      Create a list of thoughts to be drawn and their on screen locations:
    2.      Add the central thought to the list.
    3.      Add children to the list.
    4.      Add parents to the list.
    5.      Add jumps to the list.
    6.      Add siblings to the list, checking first that they are not
                already on the list.
    7.      Add distants of children to the list, checking first that they
                are not already on the list.
    8.      Add distants of parents to the list, checking first that they
                are not already on the list.
    9.      Add distants of jumps to the list, checking first that they
                are not already on the list.
    10.     Add distants of siblings to the list, checking first that
                they are not already on the list.
11.     Draw the lines that connect each thought:
    12.     For each item in the list:
        13.     Get each item in the list:
            14.     If the two items are related, draw lines
                        between them from and to the appropriate
                        gates.
15.     Draw the distant thoughts:
    16.     For each item in the list:
        17.     If it is a distant thought, draw it.
18.     Draw the other thoughts:
    19.     For each item in the list:
        20.     If it is not a distant thought, draw it.
```

FIG. 23

```
line connects the method for searching thoughts
lines to find a route from nSrc to nDest other than a direct relation
returns TRUE if found
boolean Search(int nSrc. int nDest)

{   //create the lists
    ThoughtList posList;        list of thoughts that possibly connect
    ThoughtList notList;        list of thought that do not connect
    //empty the lists
     posList.Initialize();
     notList.Initialize();
    //add the source to the not list since we cannot go directly to the destination
     notList.Add(nSrc);
    //since we cannot go directly to the destination,
    //add all relates except the destination to the possible list
     Thought src(nSrc);
     for(int n = 0; ; n++)
     {
          int nRel = src.GetRelate(n);
          if(!nRel)
          {
               //no more relations, done
               break;
          }
          if(nRel != nDest)
          {
               //add it to the possibly connect list
               posList.Add(nRel);
          }
}
while(TRUE)
{
          //check the first possibility
          int nTest = posList.GetFirst();
          if(!nTest)
          {
               //nothing on the list, done
               break;
          }
          Thought test(nTest);
          if(test.IsRelated(nDest))
          {
               //this one is related to the destination, we're done
               return TRUE:
          }
          //does not connect, add it to the does not connect list
          notList.Add(nTest);
          //add all related thoughts except those already checked to possible list
          for(int n = 0; ;++)
{
     int nRel = test.GetRelate(n);
     if(!nRel)
     {
               //no more relations, done
               break;
     }
     if(!notList.Exists(nRel))
     {
          //not checked yet, add to possible list
               posList.Add(rel);
          }
     }
     //remove this one from the possible list
       posList.Remove(nTest);
  }
    // we've checked everything there is
      no other way to get from nSrc to nDest
      return FALSE:
}
```

FIG. 24

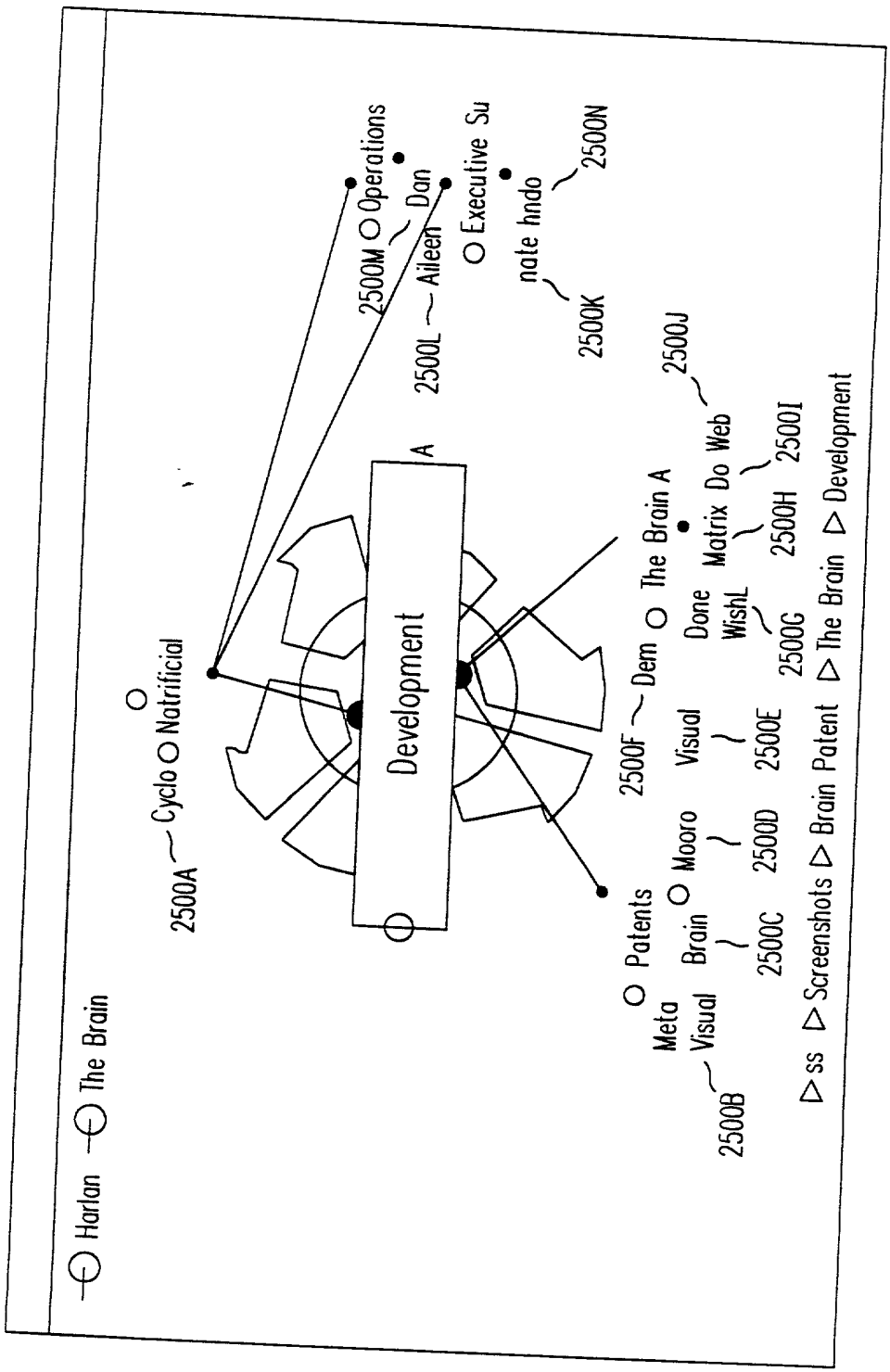

Harlan
The Brain
2500A
Cyclo
Natrificial
Development
2500M
Operations
Dan
2500L
Aileen
Executive Su
nate hndo
2500K
2500N
2500J
2500F
Dem
The Brain A
Patents
Meta
Visual
2500B
Brain
Mooro
Visual
Done
WishL
Matrix
Do Web
2500C
2500D
2500E
2500G
2500H
2500I
ss
Screenshots
Brain Patent
The Brain
Development

FIG. 25

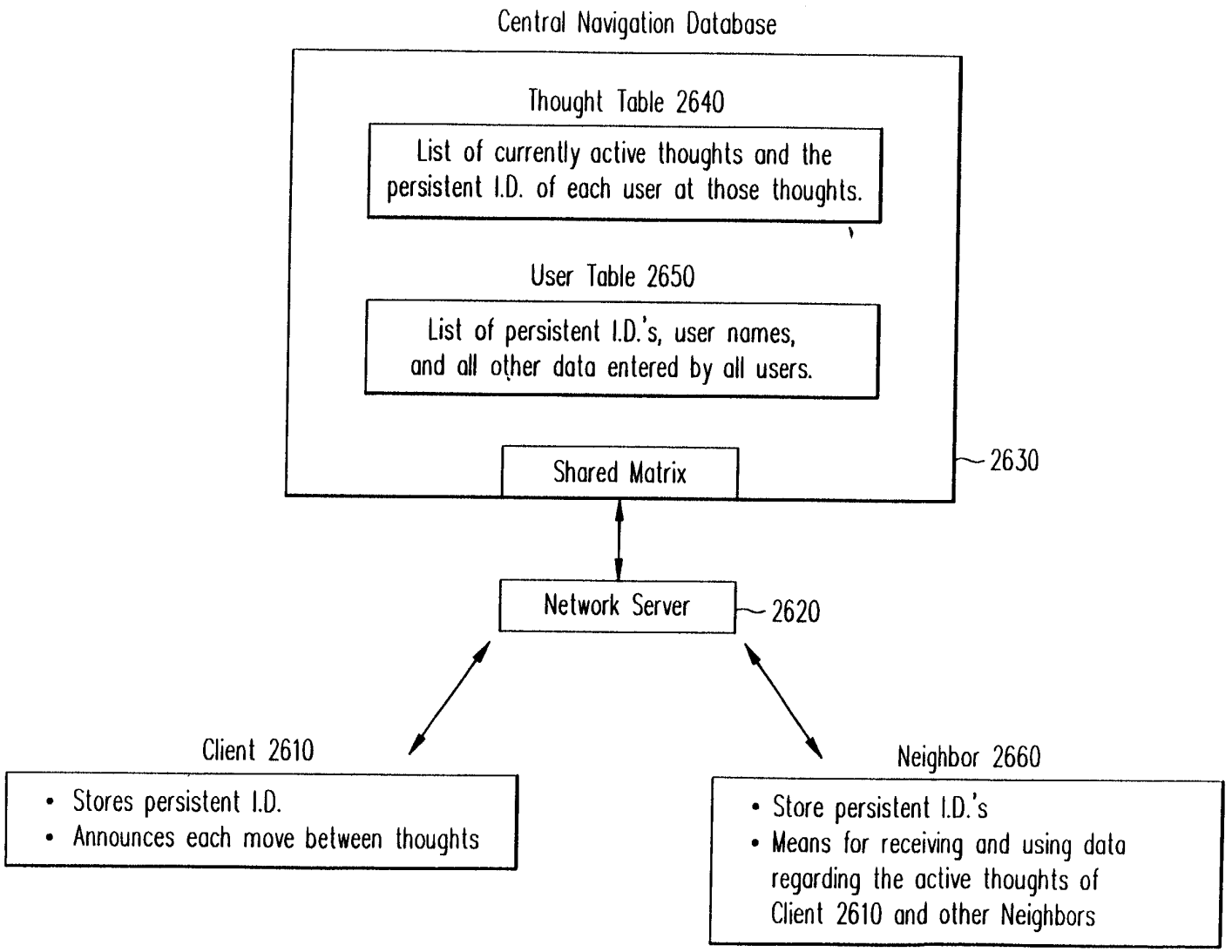

Central Navigation Database
Thought Table 2640
List of currently active thoughts and the persistent I.D. of each user at those thoughts.
User Table 2650
List of persistent I.D.'s, user names, and all other data entered by all users.
Shared Matrix
2630
Network Server
2620
Client 2610
• Stores persistent I.D.
• Announces each move between thoughts
Neighbor 2660
• Store persistent I.D.'s
• Means for receiving and using data regarding the active thoughts of Client 2610 and other Neighbors

FIG. 26

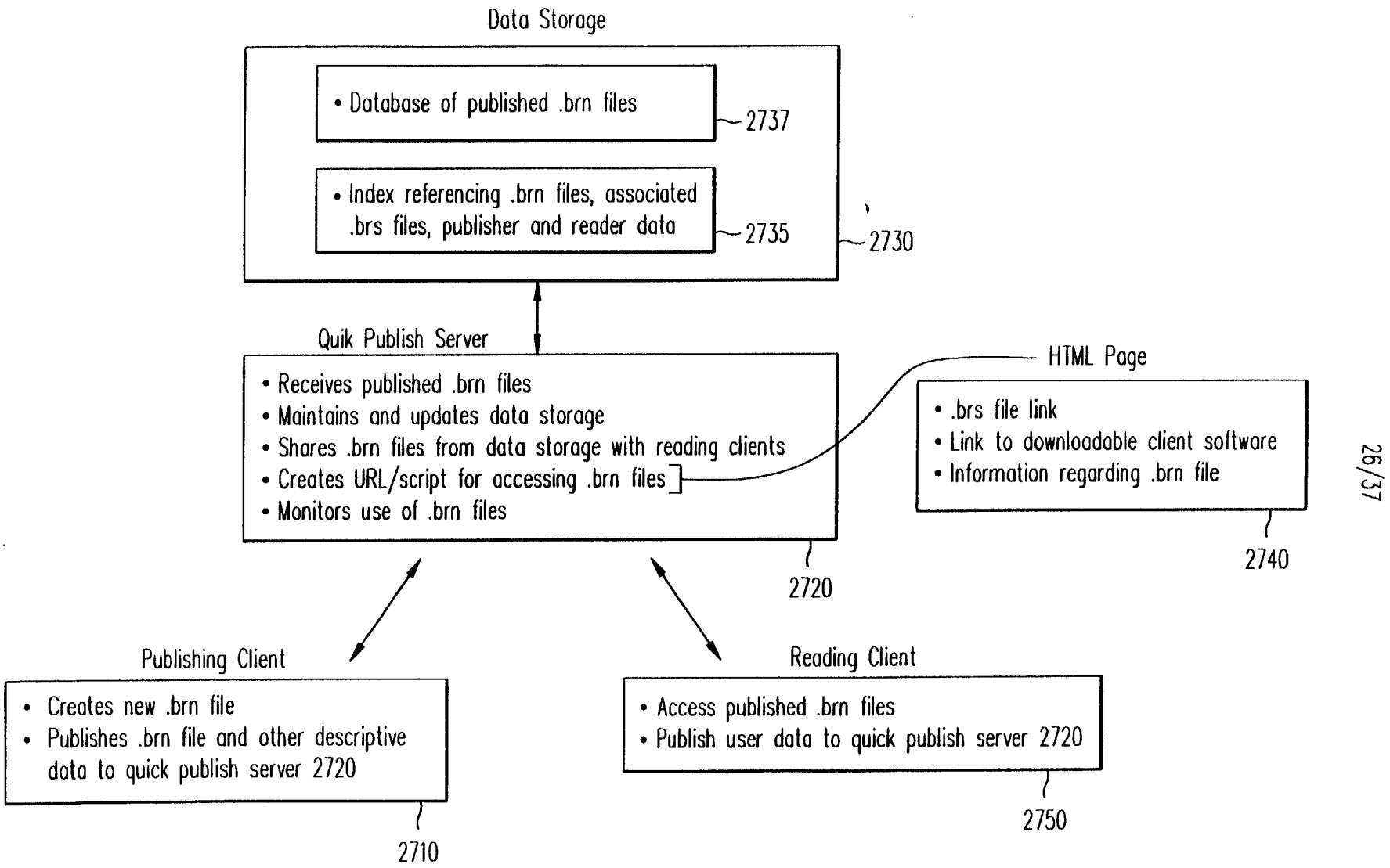

Data Storage
• Database of published .brn files
2737
• Index referencing .brn files, associated .brs files, publisher and reader data
2735
2730
Quik Publish Server
• Receives published .brn files
• Maintains and updates data storage
• Shares .brn files from data storage with reading clients
• Creates URL/script for accessing .brn files
• Monitors use of .brn files
2720
HTML Page
• .brs file link
• Link to downloadable client software
• Information regarding .brn file
2740
Publishing Client
• Creates new .brn file
• Publishes .brn file and other descriptive data to quick publish server 2720
2710
Reading Client
• Access published .brn files
• Publish user data to quick publish server 2720
2750

FIG. 27

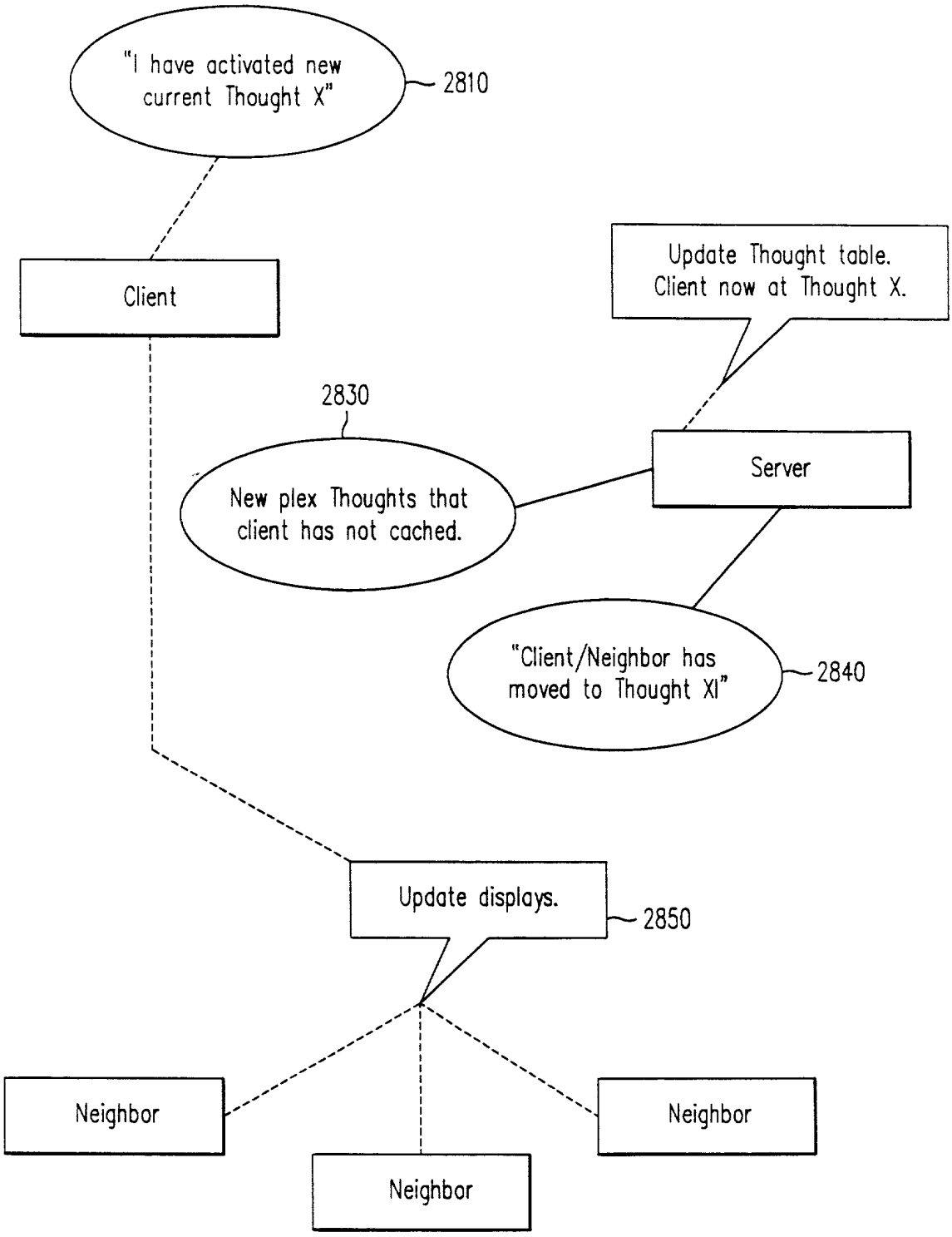
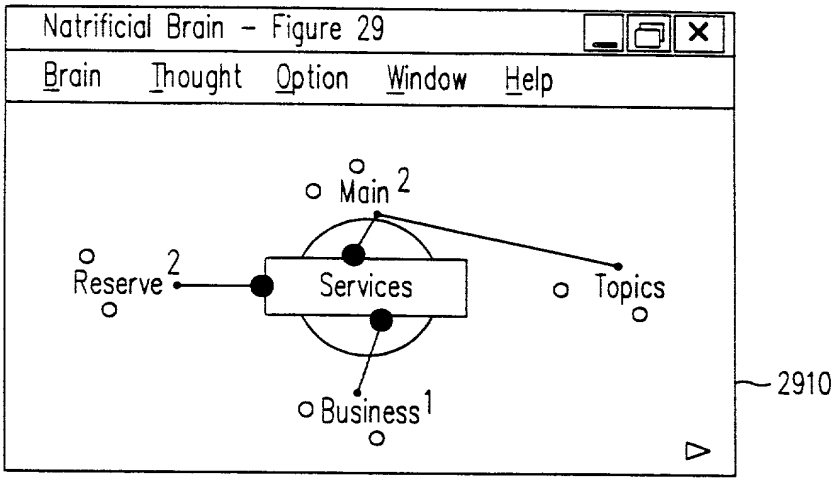
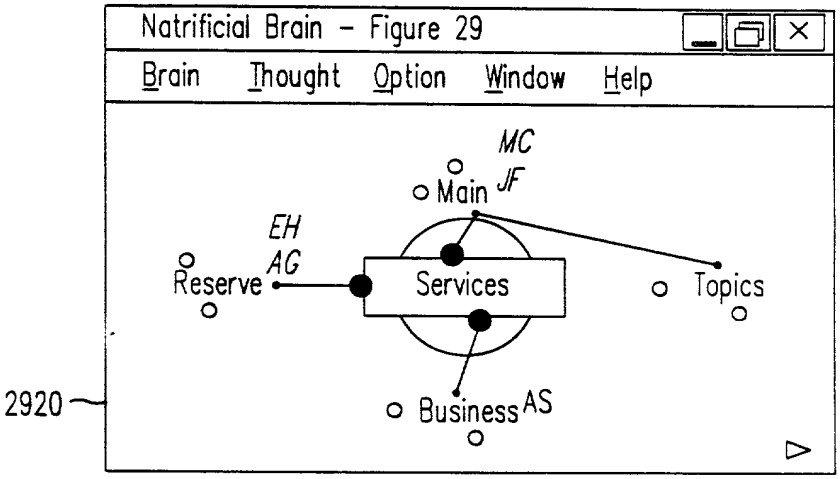

"I have activated new current Thought X"
2810
Client
Update Thought table. Client now at Thought X.
2830
New plex Thoughts that client has not cached.
Server
"Client/Neighbor has moved to Thought XI"
2840
Update displays.
2850
Neighbor
Neighbor
Neighbor

FIG. 28


Natrificial Brain – Figure 29
Brain Thought Option Window Help
Main 2
Reserve 2
Services
Topics
Business 1
2910


Natrificial Brain – Figure 29
Brain Thought Option Window Help
MC
Main JF
EH
AG
Reserve
Services
Topics
Business AS
2920

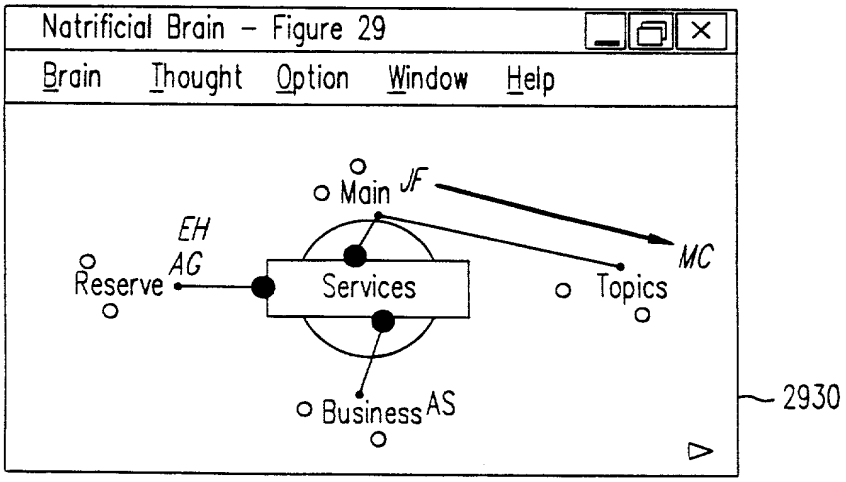

Natrificial Brain – Figure 29
Brain Thought Option Window Help
Main JF
MC
EH
AG
Reserve
Services
Topics
Business AS
2930

FIG. 29

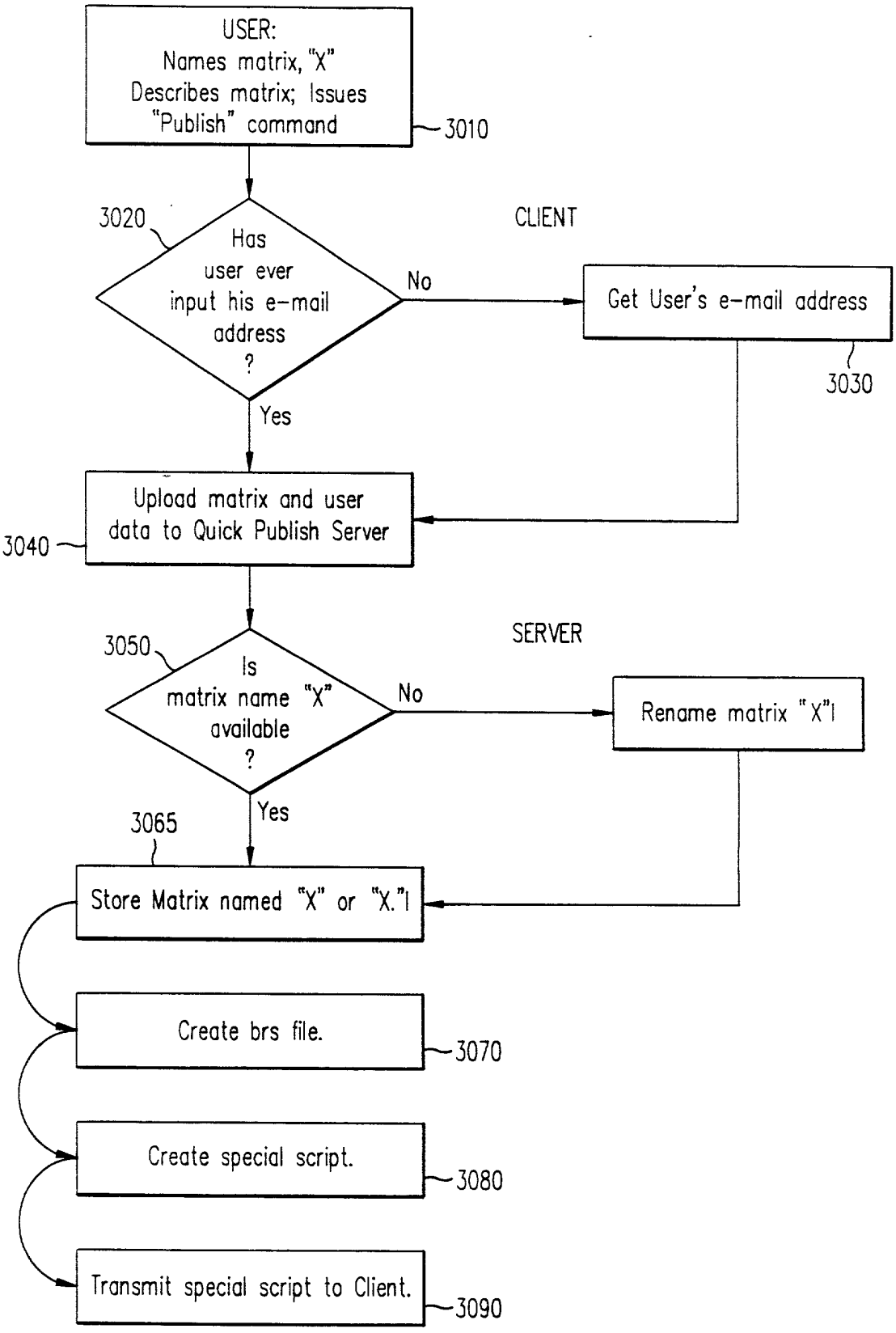

USER:
Names matrix, "X"
Describes matrix; Issues "Publish" command
3010
CLIENT
3020
Has user ever input his e-mail address ?
No
Get User's e-mail address
3030
Yes
Upload matrix and user data to Quick Publish Server
3040
SERVER
3050
Is matrix name "X" available ?
No
Rename matrix "X"!
Yes
3065
Store Matrix named "X" or "X."!
Create brs file.
3070
Create special script.
3080
Transmit special script to Client.
3090

FIG. 30

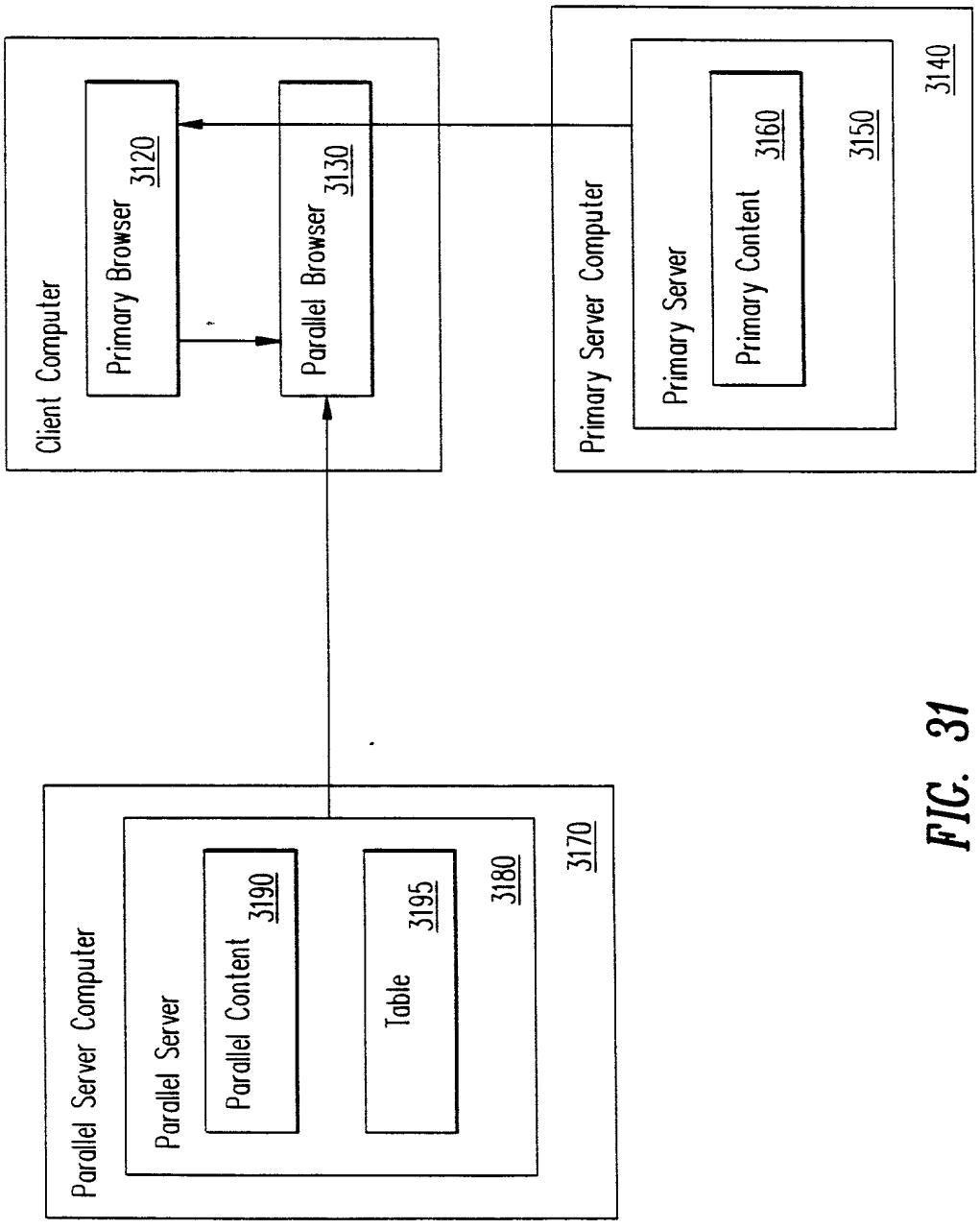

Client Computer
Primary Browser 3120
Parallel Browser 3130
Primary Server Computer
Primary Server
Primary Content 3160
3150
3140
Parallel Server Computer
Parallel Server
Parallel Content 3190
Table 3195
3180
3170

FIG. 31

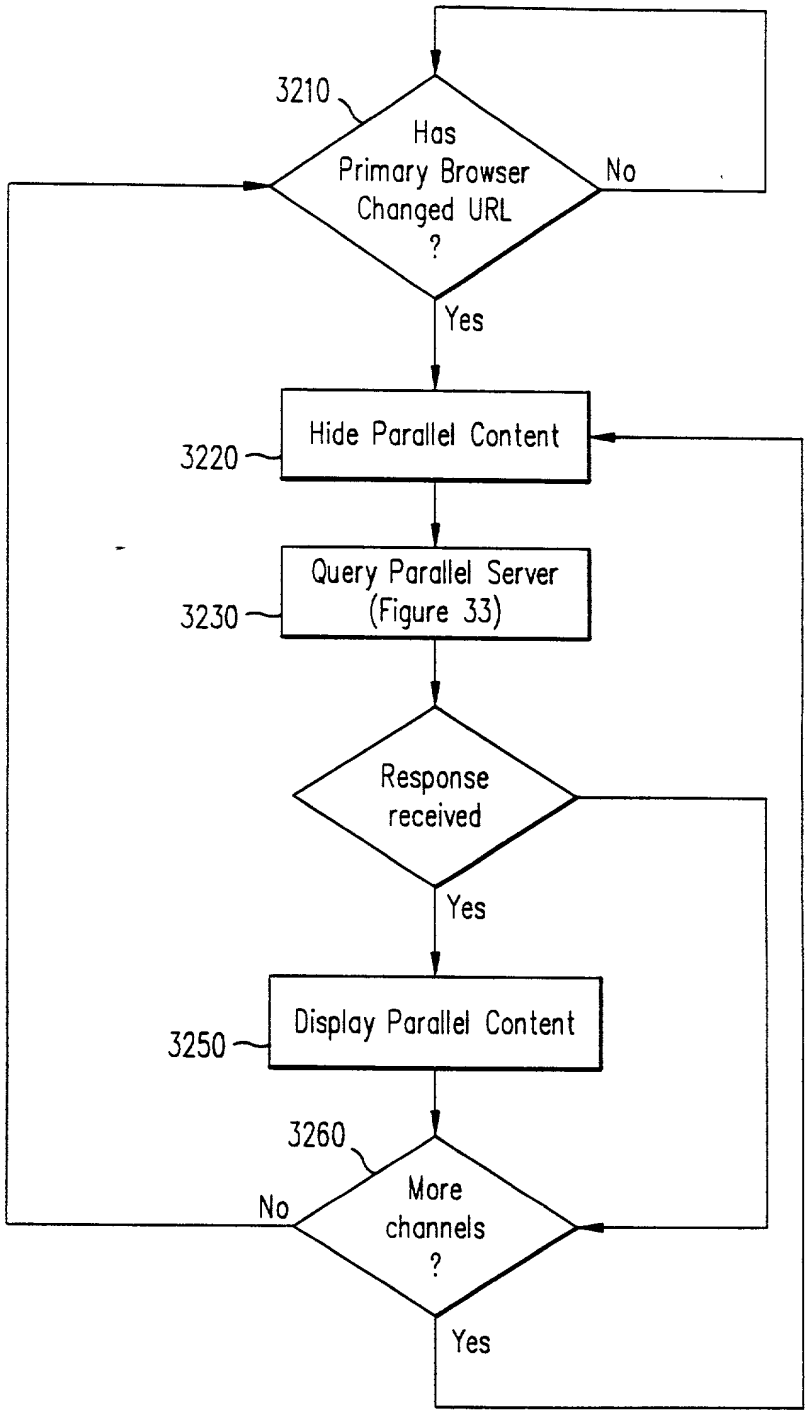

3210
Has Primary Browser Changed URL ?
No
Yes
3220
Hide Parallel Content
3230
Query Parallel Server (Figure 33)
Response received
Yes
3250
Display Parallel Content
3260
More channels ?
No
Yes

FIG. 32

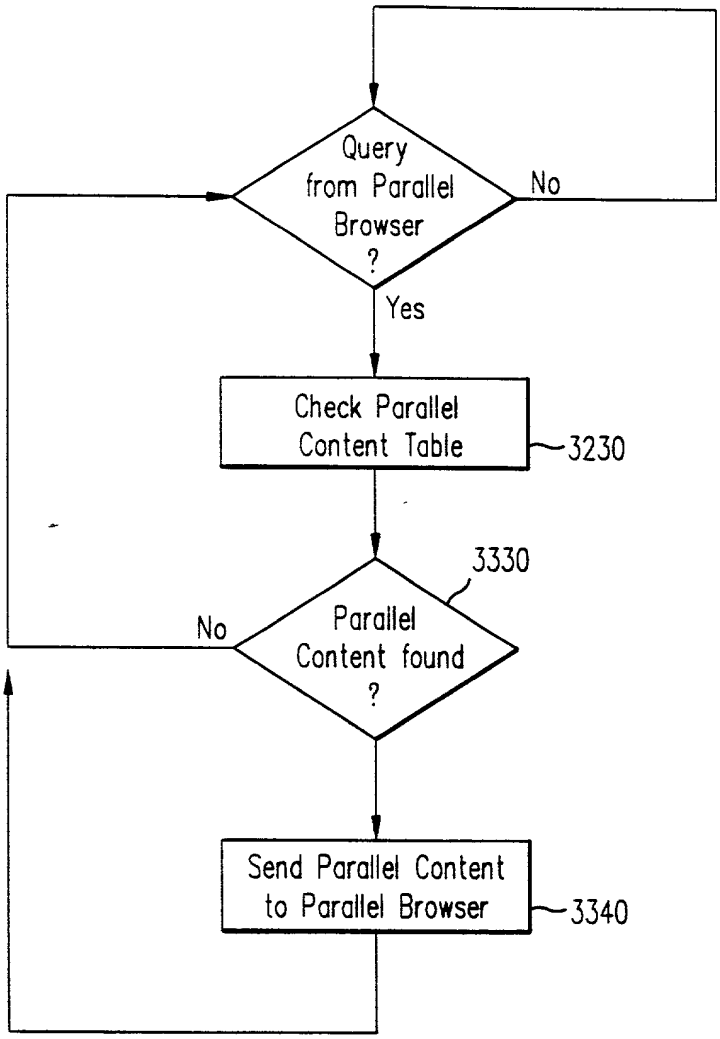
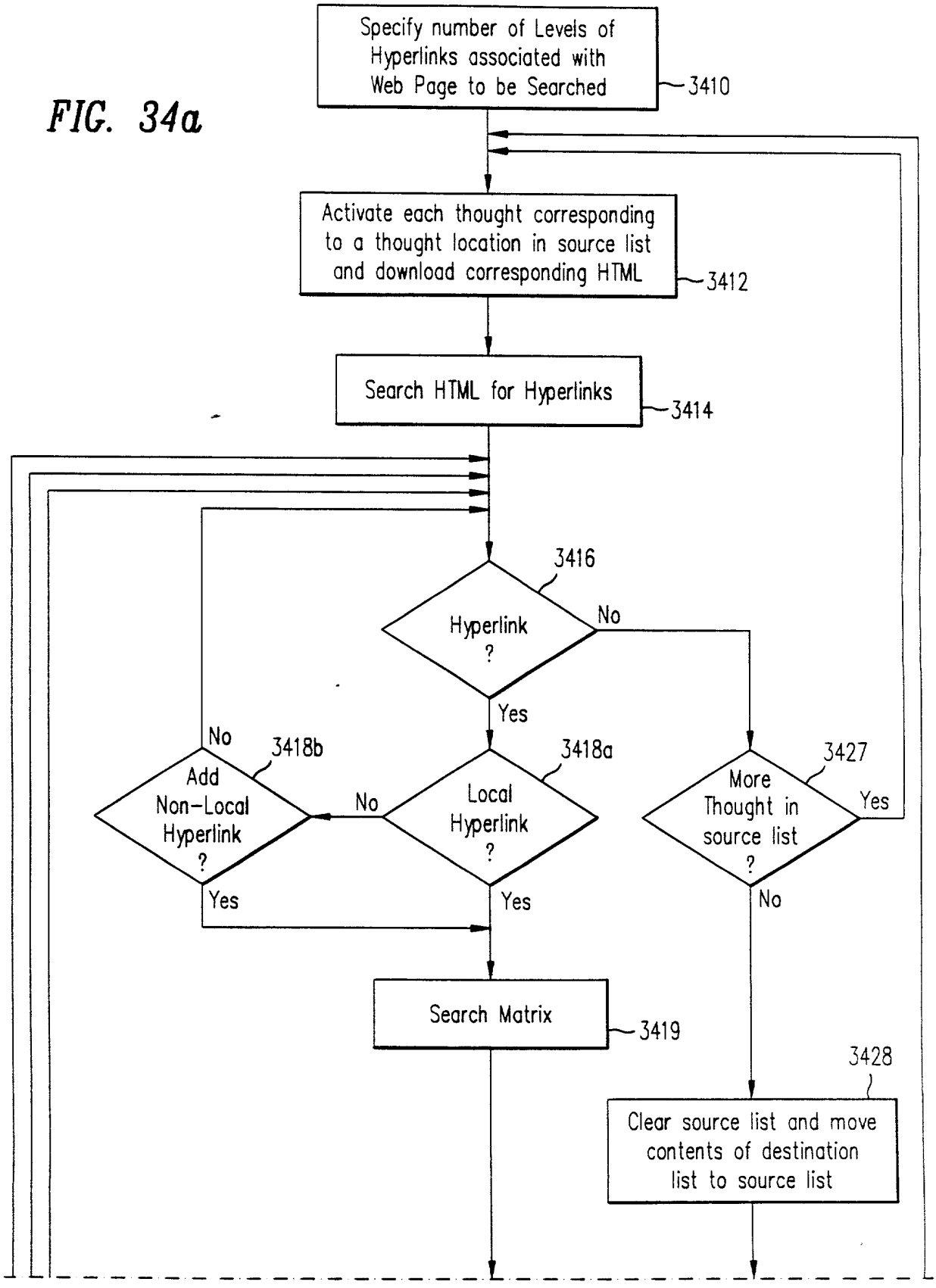

Query from Parallel Browser ?
No
Yes
Check Parallel Content Table
3230
3330
Parallel Content found ?
No
Send Parallel Content to Parallel Browser
3340

FIG. 33

FIG. 34a


Specify number of Levels of Hyperlinks associated with Web Page to be Searched
3410
Activate each thought corresponding to a thought location in source list and download corresponding HTML
3412
Search HTML for Hyperlinks
3414
3416
Hyperlink ?
No
Yes
3418b
Add Non-Local Hyperlink ?
No
Yes
3418a
Local Hyperlink ?
No
Yes
3427
More Thought in source list ?
Yes
No
Search Matrix
3419
3428
Clear source list and move contents of destination list to source list

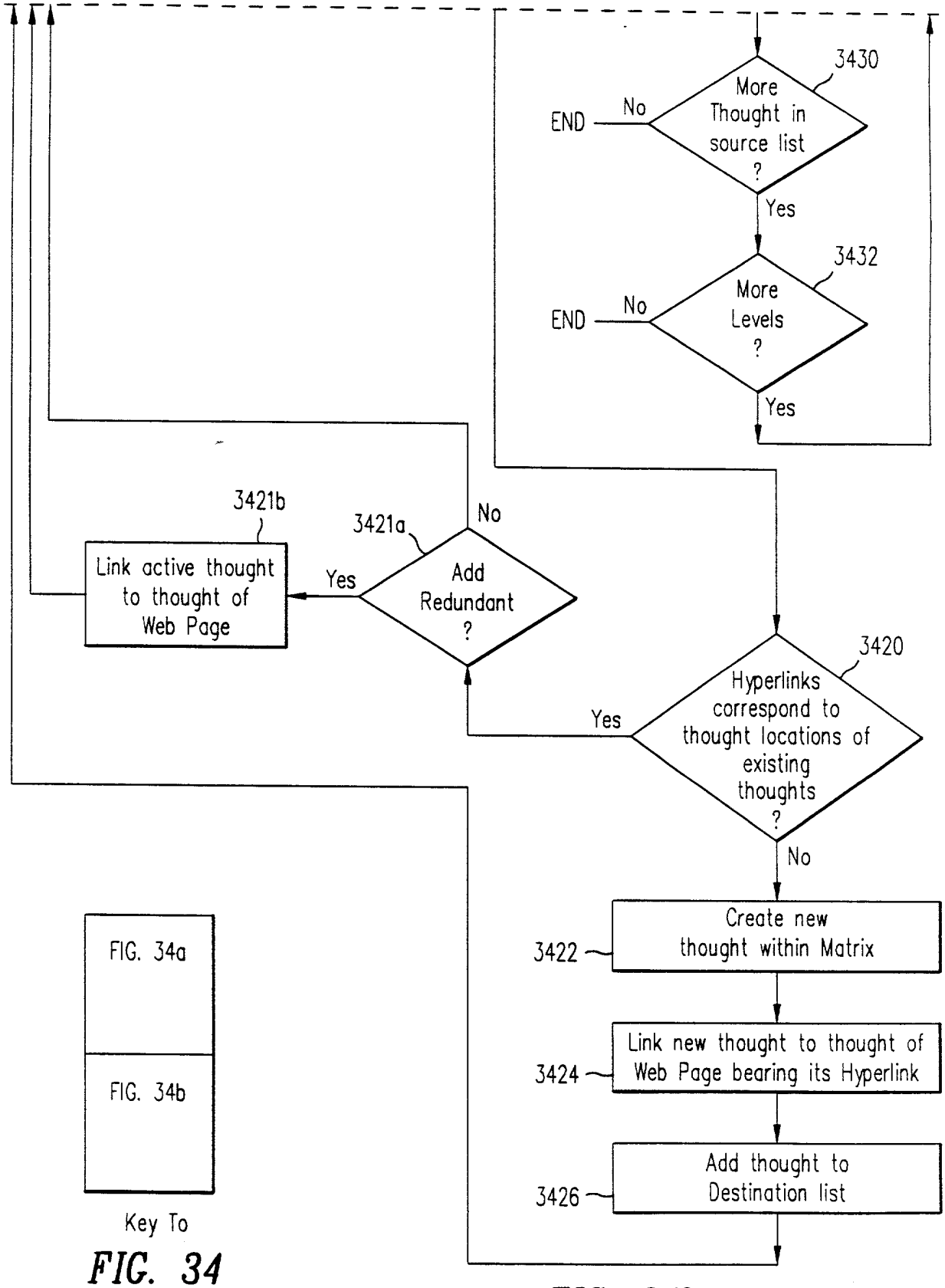


FIG. 34b

FIG. 34a

FIG. 34b

Key To

FIG. 34

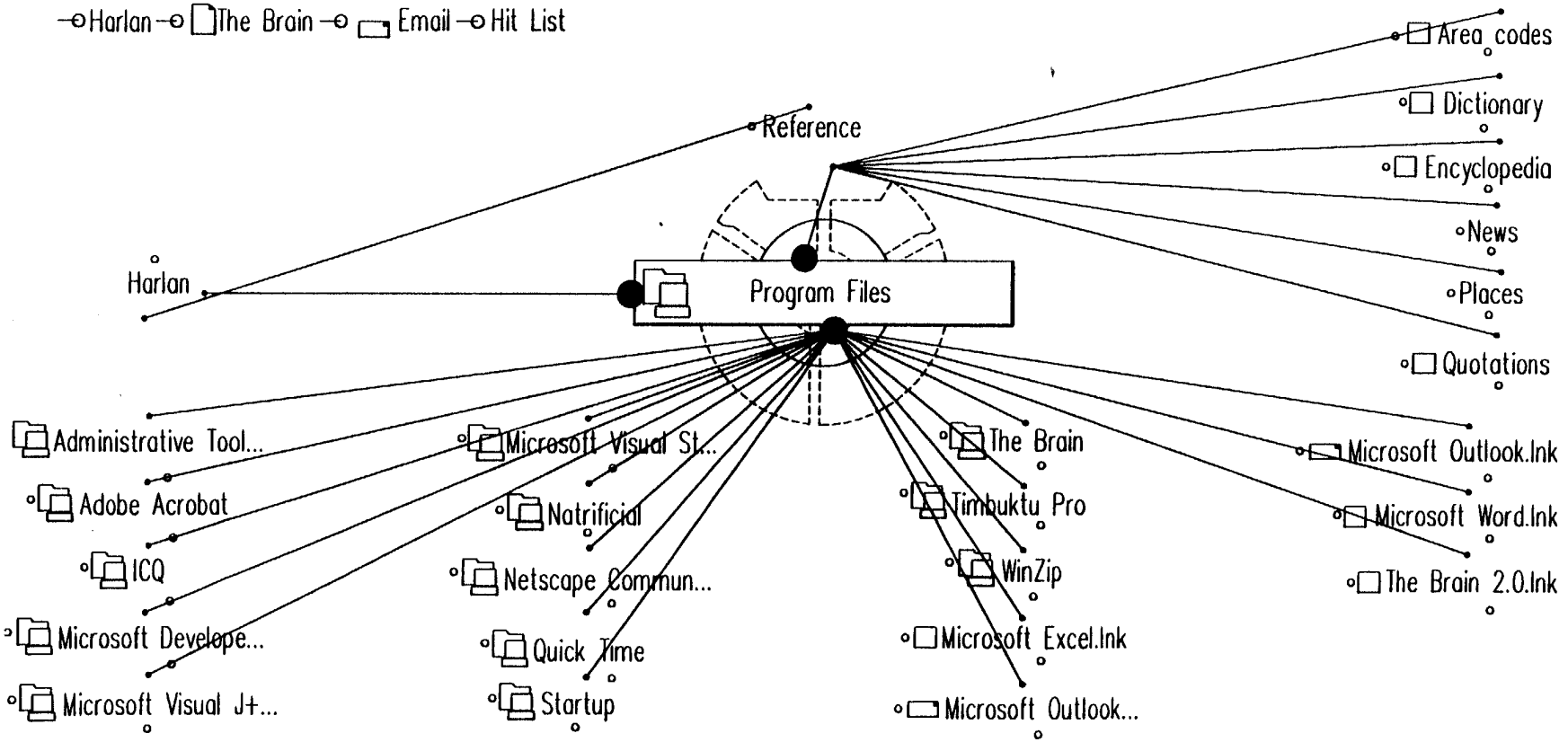

Harlan
The Brain
Email
Hit List
Reference
Harlan
Program Files
Area codes
Dictionary
Encyclopedia
News
Places
Quotations
Administrative Tool...
Adobe Acrobat
ICQ
Microsoft Develope...
Microsoft Visual J+...
Microsoft Visual St...
Natrificial
Netscape Commun...
Quick Time
Startup
The Brain
Timbuktu Pro
WinZip
Microsoft Excel.lnk
Microsoft Outlook...
Microsoft Outlook.lnk
Microsoft Word.lnk
The Brain 2.0.lnk

FIG. 36

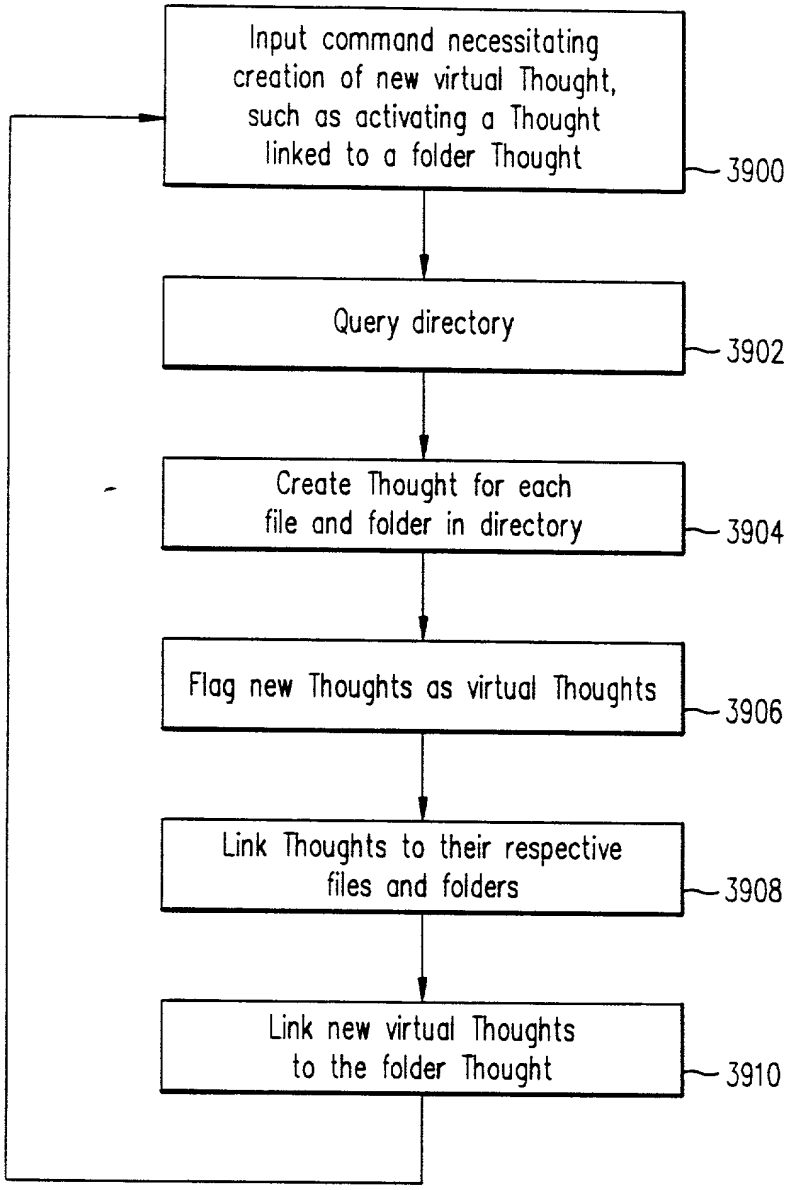

Input command necessitating creation of new virtual Thought, such as activating a Thought linked to a folder Thought
3900
Query directory
3902
Create Thought for each file and folder in directory
3904
Flag new Thoughts as virtual Thoughts
3906
Link Thoughts to their respective files and folders
3908
Link new virtual Thoughts to the folder Thought
3910

FIG. 37

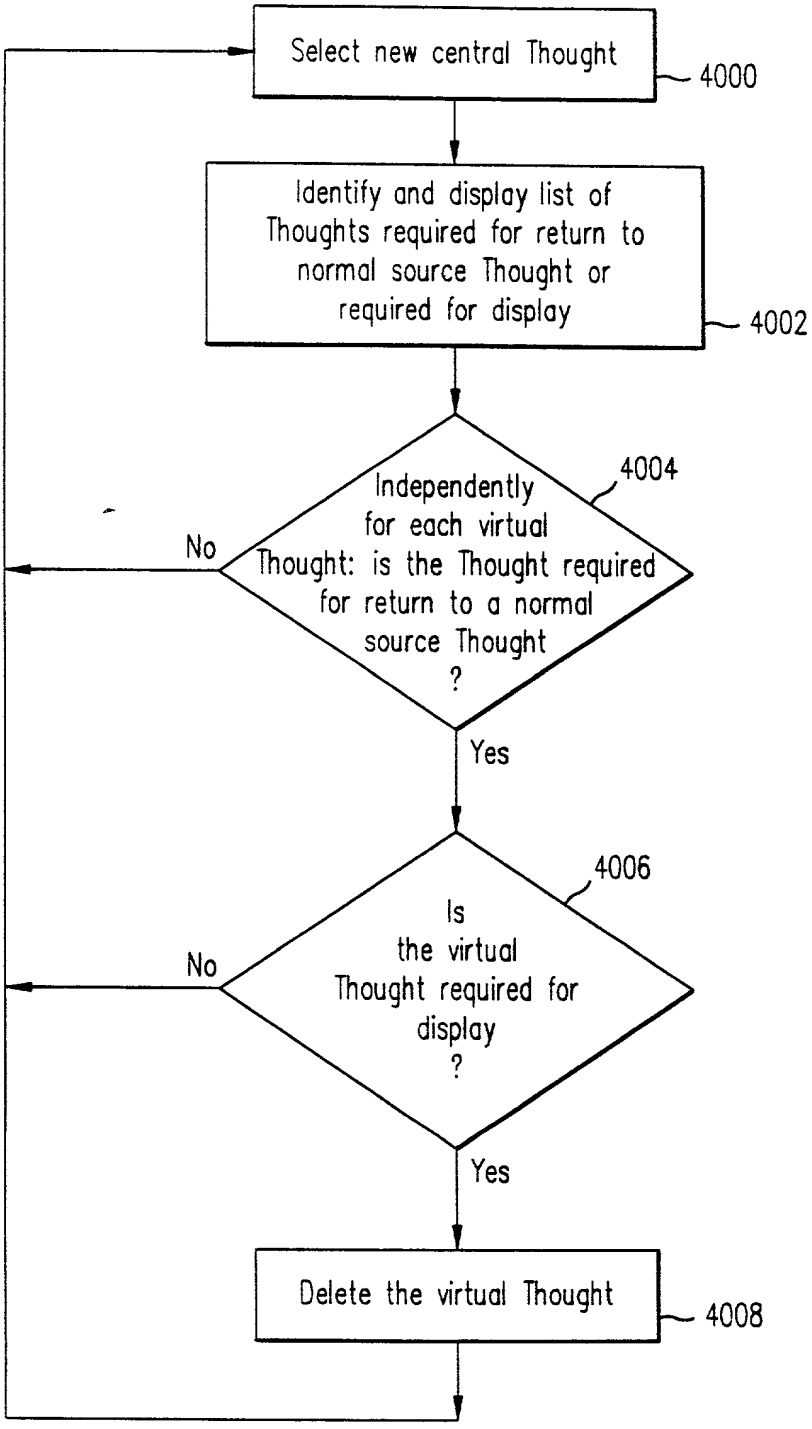

Select new central Thought
4000
Identify and display list of Thoughts required for return to normal source Thought or required for display
4002
Independently for each virtual Thought: is the Thought required for return to a normal source Thought ?
4004
No
Yes
Is the virtual Thought required for display ?
4006
No
Yes
Delete the virtual Thought
4008

FIG. 38